



Consolidated Platform Configuration Guide, Cisco IOS 15.2(6)E (Catalyst 2960-XR Switch)

First Published: 2017-08-07

Americas Headquarters

Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA 95134-1706
USA
<http://www.cisco.com>
Tel: 408 526-4000
800 553-NETS (6387)
Fax: 408 527-0883

THE SPECIFICATIONS AND INFORMATION REGARDING THE PRODUCTS IN THIS MANUAL ARE SUBJECT TO CHANGE WITHOUT NOTICE. ALL STATEMENTS, INFORMATION, AND RECOMMENDATIONS IN THIS MANUAL ARE BELIEVED TO BE ACCURATE BUT ARE PRESENTED WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED. USERS MUST TAKE FULL RESPONSIBILITY FOR THEIR APPLICATION OF ANY PRODUCTS.

THE SOFTWARE LICENSE AND LIMITED WARRANTY FOR THE ACCOMPANYING PRODUCT ARE SET FORTH IN THE INFORMATION PACKET THAT SHIPPED WITH THE PRODUCT AND ARE INCORPORATED HEREIN BY THIS REFERENCE. IF YOU ARE UNABLE TO LOCATE THE SOFTWARE LICENSE OR LIMITED WARRANTY, CONTACT YOUR CISCO REPRESENTATIVE FOR A COPY.

The Cisco implementation of TCP header compression is an adaptation of a program developed by the University of California, Berkeley (UCB) as part of UCB's public domain version of the UNIX operating system. All rights reserved. Copyright © 1981, Regents of the University of California.

NOTWITHSTANDING ANY OTHER WARRANTY HEREIN, ALL DOCUMENT FILES AND SOFTWARE OF THESE SUPPLIERS ARE PROVIDED "AS IS" WITH ALL FAULTS. CISCO AND THE ABOVE-NAMED SUPPLIERS DISCLAIM ALL WARRANTIES, EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, THOSE OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NON-INFRINGEMENT OR ARISING FROM A COURSE OF DEALING, USAGE, OR TRADE PRACTICE.

IN NO EVENT SHALL CISCO OR ITS SUPPLIERS BE LIABLE FOR ANY INDIRECT, SPECIAL, CONSEQUENTIAL, OR INCIDENTAL DAMAGES, INCLUDING, WITHOUT LIMITATION, LOST PROFITS OR LOSS OR DAMAGE TO DATA ARISING OUT OF THE USE OR INABILITY TO USE THIS MANUAL, EVEN IF CISCO OR ITS SUPPLIERS HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

Any Internet Protocol (IP) addresses and phone numbers used in this document are not intended to be actual addresses and phone numbers. Any examples, command display output, network topology diagrams, and other figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses or phone numbers in illustrative content is unintentional and coincidental.

All printed copies and duplicate soft copies of this document are considered uncontrolled. See the current online version for the latest version.

Cisco has more than 200 offices worldwide. Addresses and phone numbers are listed on the Cisco website at www.cisco.com/go/offices.

Cisco and the Cisco logo are trademarks or registered trademarks of Cisco and/or its affiliates in the U.S. and other countries. To view a list of Cisco trademarks, go to this URL: <https://www.cisco.com/c/en/us/about/legal/trademarks.html>. Third-party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1721R)

© 2017 Cisco Systems, Inc. All rights reserved.



CONTENTS

PART I

Interface and Hardware Component 93

CHAPTER 1

Configuring Interface Characteristics 1

Information About Configuring Interface Characteristics 1

Interface Types 1

Port-Based VLANs 1

Switch Ports 2

Routed Ports 3

Switch Virtual Interfaces 4

EtherChannel Port Groups 5

Power over Ethernet Ports 5

Using the Switch USB Ports 5

USB Mini-Type B Console Port 5

USB Type A Ports 6

Interface Connections 7

Interface Configuration Mode 7

Default Ethernet Interface Configuration 8

Interface Speed and Duplex Mode 9

Speed and Duplex Configuration Guidelines 10

IEEE 802.3x Flow Control 10

Layer 3 Interfaces 11

How to Configure Interface Characteristics 12

Configuring Interfaces 12

Adding a Description for an Interface 13

Configuring a Range of Interfaces 14

Configuring and Using Interface Range Macros 16

Configuring Ethernet Interfaces	17
Setting the Interface Speed and Duplex Parameters	17
Configuring IEEE 802.3x Flow Control	19
Configuring Layer 3 Interfaces	20
Configuring SVI Autostate Exclude	21
Shutting Down and Restarting the Interface	22
Configuring the Console Media Type	24
Configuring the USB Inactivity Timeout	25
Monitoring Interface Characteristics	26
Monitoring Interface Status	26
Clearing and Resetting Interfaces and Counters	27
Configuration Examples for Interface Characteristics	27
Configuring a Range of Interfaces: Examples	27
Configuring and Using Interface Range Macros: Examples	28
Setting Interface Speed and Duplex Mode: Example	28
Configuring Layer 3 Interfaces: Example	29
Configuring the Console Media Type: Example	29
Configuring the USB Inactivity Timeout: Example	29
Additional References for the Interface Characteristics Feature	30
Feature History and Information for Configuring Interface Characteristics	30

CHAPTER 2

Configuring Auto-MDIX	31
Prerequisites for Auto-MDIX	31
Restrictions for Auto-MDIX	31
Information About Configuring Auto-MDIX	32
Auto-MDIX on an Interface	32
How to Configure Auto-MDIX	32
Configuring Auto-MDIX on an Interface	32
Example for Configuring Auto-MDIX	33
Additional References	34
Feature History and Information for Auto-MDIX	34

CHAPTER 3

Configuring Ethernet Management Port	35
Finding Feature Information	35

Prerequisites for Ethernet Management Ports	35
Information About the Ethernet Management Port	35
Ethernet Management Port Direct Connection to a Device	36
Ethernet Management Port Connection to Stack Devices using a Hub	36
Ethernet Management Port and Routing	36
Supported Features on the Ethernet Management Port	37
How to Configure the Ethernet Management Port	38
Disabling and Enabling the Ethernet Management Port	38
Additional References for Ethernet Management Ports	39
Feature History and Information for Ethernet Management Ports	39
<hr/>	
CHAPTER 4	Configuring LLDP, LLDP-MED, and Wired Location Service
	41
Information About LLDP, LLDP-MED, and Wired Location Service	41
LLDP	41
LLDP Supported TLVs	41
LLDP and Cisco Device Stacks	42
LLDP and Cisco Medianet	42
LLDP-MED	42
LLDP-MED Supported TLVs	42
Wired Location Service	43
Default LLDP Configuration	44
Restrictions for LLDP	45
How to Configure LLDP, LLDP-MED, and Wired Location Service	45
Enabling LLDP	45
Configuring LLDP Characteristics	47
Configuring LLDP-MED TLVs	49
Configuring Network-Policy TLV	50
Configuring Location TLV and Wired Location Service	52
Enabling Wired Location Service on the Device	55
Configuration Examples for LLDP, LLDP-MED, and Wired Location Service	56
Configuring Network-Policy TLV: Examples	56
Monitoring and Maintaining LLDP, LLDP-MED, and Wired Location Service	57
Additional References for LLDP, LLDP-MED, and Wired Location Service	58
Feature Information for LLDP, LLDP-MED, and Wired Location Service	58

CHAPTER 5	Configuring System MTU	59
	Information About the MTU	59
	Restrictions for System MTU	59
	How to Configure MTU	59
	Configuring the System MTU	59
	Configuration Examples for System MTU	60
	Additional References for System MTU	61
	Feature Information for System MTU	61

CHAPTER 6	Configuring Internal Power Supplies	63
	Information About Internal Power Supplies	63
	How to Configure Internal Power Supplies	63
	Configuring Internal Power Supply	63
	Monitoring Internal Power Supplies	64
	Configuration Examples for Internal Power Supplies	64
	Additional References	65
	Feature History and Information for Internal Power Supplies	65

CHAPTER 7	Configuring Boot Fast	67
	Configuring Boot Fast on the switch	67
	Enabling Boot Fast	67
	Disabling Boot Fast	68

CHAPTER 8	Configuring PoE	71
	Information About PoE	71
	Power over Ethernet Ports	71
	Supported Protocols and Standards	71
	Powered-Device Detection and Initial Power Allocation	72
	Power Management Modes	73
	How to Configure PoE	76
	Configuring a Power Management Mode on a PoE Port	76
	Budgeting Power for Devices Connected to a PoE Port	77
	Budgeting Power to All PoE ports	78

	Budgeting Power to a Specific PoE Port	79
	Configuring Power Policing	80
	Monitoring Power Status	83
	Configuration Examples for Configuring PoE	83
	Budgeting Power: Example	83
	Additional References	83
<hr/>		
CHAPTER 9	Configuring 2-event Classification	85
	Information about 2-event Classification	85
	Configuring 2-event Classification	85
	Example: Configuring 2-Event Classification	86
<hr/>		
CHAPTER 10	Configuring EEE	87
	Restrictions for EEE	87
	Information About EEE	87
	EEE Overview	87
	Default EEE Configuration	87
	How to Configure EEE	88
	Enabling or Disabling EEE	88
	Monitoring EEE	89
	Configuration Examples for Configuring EEE	90
	Additional References	90
	Feature Information for Configuring EEE	91
<hr/>		
PART II	IP Multicast Routing	93
<hr/>		
CHAPTER 11	IP Multicast Routing Technology Overview	95
	Information About IP Multicast Technology	95
	Role of IP Multicast in Information Delivery	95
	IP Multicast Routing Protocols	95
	Multicast Group Transmission Scheme	96
	IP Multicast Boundary	97
	IP Multicast Group Addressing	97
	IP Class D Addresses	98

IP Multicast Address Scoping	98
Layer 2 Multicast Addresses	99
IP Multicast Delivery Modes	100
Source Specific Multicast	100
Additional References	100

CHAPTER 12**Configuring Basic IP Multicast Routing 103**

Prerequisites for Basic IP Multicast Routing	103
Restrictions for Basic IP Multicast Routing	103
Information About Basic IP Multicast Routing	104
Multicast Routing and Device Stacks	104
Default IP Multicast Routing Configuration	104
Configuring sdr Listener Support	105
How to Configure Basic IP Multicast Routing	105
Configuring Basic IP Multicast Routing	105
Configuring Optional IP Multicast Routing Features	108
Defining the IP Multicast Boundary	108
Configuring Multicast VRFs	109
Advertising Multicast Multimedia Sessions Using SAP Listener	111
Monitoring and Maintaining Basic IP Multicast Routing	112
Clearing Caches, Tables, and Databases	112
Displaying System and Network Statistics	113
Displaying Multicast Peers, Packet Rates and Loss Information, and Path Tracing	114
Additional References	115
Feature History and Information for IP Multicast	116

CHAPTER 13**Configuring IGMP 117**

Prerequisites for IGMP	117
Restrictions for Configuring IGMP	117
Information About IGMP	118
Role of the Internet Group Management Protocol	118
IGMP Multicast Addresses	118
IGMP Versions	118
IGMP Version 1	119

IGMP Version 2	119
IGMP Version 3	119
IGMPv3 Host Signalling	119
IGMP Versions Differences	119
IGMP Join and Leave Process	121
IGMP Join Process	121
IGMP Leave Process	122
Default IGMP Configuration	122
How to Configure IGMP	123
Configuring the Device as a Member of a Group	123
Controlling Access to IP Multicast Group	124
Changing the IGMP Version	126
Modifying the IGMP Host-Query Message Interval	128
Changing the IGMP Query Timeout for IGMPv2	129
Changing the Maximum Query Response Time for IGMPv2	131
Configuring the Device as a Statically Connected Member	132
Monitoring IGMP	134
Configuration Examples for IGMP	135
Example: Configuring the Device as a Member of a Multicast Group	135
Example: Controlling Access to IP Multicast Groups	135
Additional References	135
Feature History and Information for IGMP	136

CHAPTER 14

Configuring IGMP Snooping and Multicast VLAN Registration	137
Prerequisites for Configuring IGMP Snooping and MVR	137
Prerequisites for IGMP Snooping	137
Restrictions for Configuring IGMP Snooping and MVR	138
Restrictions for IGMP Snooping	138
Restrictions for MVR	138
Information About IGMP Snooping and MVR	139
IGMP Snooping	139
IGMP Versions	140
Joining a Multicast Group	140
Leaving a Multicast Group	142

Immediate Leave	142
IGMP Configurable-Leave Timer	142
IGMP Report Suppression	142
IGMP Snooping and Device Stacks	143
Default IGMP Snooping Configuration	143
Multicast VLAN Registration	143
MVR and IGMP	144
Modes of Operation	144
MVR and Switch Stacks	144
MVR in a Multicast Television Application	144
Default MVR Configuration	146
IGMP Filtering and Throttling	146
Default IGMP Filtering and Throttling Configuration	147
How to Configure IGMP Snooping and MVR	147
Enabling or Disabling IGMP Snooping on a Device	147
Enabling or Disabling IGMP Snooping on a VLAN Interface	148
Setting the Snooping Method	149
Configuring a Multicast Router Port	151
Configuring a Host Statically to Join a Group	152
Enabling IGMP Immediate Leave	154
Configuring the IGMP Leave Timer	155
Configuring TCN-Related Commands	156
Controlling the Multicast Flooding Time After a TCN Event	156
Recovering from Flood Mode	158
Disabling Multicast Flooding During a TCN Event	159
Configuring the IGMP Snooping Querier	160
Disabling IGMP Report Suppression	162
Configuring MVR Global Parameters	163
Configuring MVR Interfaces	166
Configuring IGMP Profiles	168
Applying IGMP Profiles	170
Setting the Maximum Number of IGMP Groups	171
Configuring the IGMP Throttling Action	173
Monitoring IGMP Snooping and MVR	175

Monitoring IGMP Snooping Information	175
Monitoring MVR	176
Monitoring IGMP Filtering and Throttling Configuration	177
Configuration Examples for IGMP Snooping and MVR	178
Example: Configuring IGMP Snooping Using CGMP Packets	178
Example: Enabling a Static Connection to a Multicast Router	178
Example: Configuring a Host Statically to Join a Group	178
Example: Enabling IGMP Immediate Leave	178
Example: Setting the IGMP Snooping Querier Source Address	178
Example: Setting the IGMP Snooping Querier Maximum Response Time	178
Example: Setting the IGMP Snooping Querier Timeout	179
Example: Setting the IGMP Snooping Querier Feature	179
Example: Configuring IGMP Profiles	179
Example: Applying IGMP Profile	179
Example: Setting the Maximum Number of IGMP Groups	179
Example: Configuring MVR Global Parameters	180
Example: Configuring MVR Interfaces	180
Additional References	180
Feature History and Information for IGMP Snooping	181

CHAPTER 15
Configuring CGMP 183

Finding Feature Information	183
Prerequisites for Configuring CGMP	183
Restrictions for CGMP	183
Information About CGMP	184
Enabling CGMP Server Support	184
Monitoring CGMP	186
Additional References	187
Feature History and Information for CGMP	188

CHAPTER 16
Configuring Protocol Independent Multicast (PIM) 189

Prerequisites for PIM	189
Restrictions for PIM	190
PIMv1 and PIMv2 Interoperability	190

Restrictions for Configuring PIM Stub Routing	190
Restrictions for Configuring Auto-RP and BSR	191
Information About PIM	192
Protocol Independent Multicast Overview	192
PIM Dense Mode	192
PIM Sparse Mode	193
Sparse-Dense Mode	193
PIM Versions	194
PIM Stub Routing	194
IGMP Helper	196
Rendezvous Points	196
Auto-RP	196
Sparse-Dense Mode for Auto-RP	197
Bootstrap Router	198
PIM Domain Border	198
Multicast Forwarding	198
Multicast Distribution Source Tree	198
Multicast Distribution Shared Tree	199
Source Tree Advantage	200
Shared Tree Advantage	200
PIM Shared Tree and Source Tree	201
Reverse Path Forwarding	202
RPF Check	203
Default PIM Routing Configuration	204
How to Configure PIM	205
Enabling PIM Stub Routing	205
Configuring a Rendezvous Point	206
Manually Assigning an RP to Multicast Groups	207
Setting Up Auto-RP in a New Internetwork	209
Adding Auto-RP to an Existing Sparse-Mode Cloud	212
Configuring Sparse Mode with a Single Static RP(CLI)	215
Preventing Join Messages to False RPs	217
Filtering Incoming RP Announcement Messages	217
Configuring PIMv2 BSR	219

Defining the PIM Domain Border	219
Defining the IP Multicast Boundary	221
Configuring Candidate BSRs	222
Configuring the Candidate RPs	224
Delaying the Use of PIM Shortest-Path Tree	226
Modifying the PIM Router-Query Message Interval	228
Verifying PIM Operations	229
Verifying IP Multicast Operation in a PIM-SM or a PIM-SSM Network	229
Using PIM-Enabled Routers to Test IP Multicast Reachability	235
Monitoring and Troubleshooting PIM	237
Monitoring PIM Information	237
Monitoring the RP Mapping and BSR Information	238
Troubleshooting PIMv1 and PIMv2 Interoperability Problems	238
Configuration Examples for PIM	239
Example: Enabling PIM Stub Routing	239
Example: Verifying PIM Stub Routing	239
Example: Manually Assigning an RP to Multicast Groups	240
Example: Configuring Auto-RP	240
Example: Defining the IP Multicast Boundary to Deny Auto-RP Information	240
Example: Filtering Incoming RP Announcement Messages	240
Example: Preventing Join Messages to False RPs	241
Example: Configuring Candidate BSRs	241
Example: Configuring Candidate RPs	241
Additional References	241
Feature History and Information for PIM	243

CHAPTER 17**Configuring HSRP Aware PIM 245**

HSRP Aware PIM	245
Restrictions for HSRP Aware PIM	245
Information About HSRP Aware PIM	245
HSRP	245
HSRP Aware PIM	246
How to Configure HSRP Aware PIM	247
Configuring an HSRP Group on an Interface	247

Configuring PIM Redundancy	249
Configuration Examples for HSRP Aware PIM	250
Example: Configuring an HSRP Group on an Interface	250
Example: Configuring PIM Redundancy	250
Additional References for HSRP Aware PIM	250
Feature Information for HSRP Aware PIM	251

CHAPTER 18**Configuring VRRP Aware PIM 253**

VRRP Aware PIM	253
Restrictions for VRRP Aware PIM	253
Information About VRRP Aware PIM	253
Overview of VRRP Aware PIM	253
How to Configure VRRP Aware PIM	254
Configuring VRRP Aware PIM	254
Configuration Examples for VRRP Aware PIM	256
Example: VRRP Aware PIM	256
Additional References for VRRP Aware PIM	256

CHAPTER 19**Configuring SSM 259**

Prerequisites for Configuring SSM	259
Restrictions for Configuring SSM	259
Information About SSM and SSM Mapping	261
SSM Components Overview	261
How SSM Differs from Internet Standard Multicast	261
SSM Operations	262
IGMPv3 Host Signaling	262
Benefits of Source Specific Multicast	263
SSM Mapping Overview	264
Static SSM Mapping	264
DNS-Based SSM Mapping	264
SSM Mapping Benefits	266
How to Configure SSM and SSM Mapping	266
Configuring SSM	266
Configuring SSM Mapping	268

Configuring Static SSM Mapping	268
Configuring DNS-Based SSM Mapping (CLI)	269
Configuring Static Traffic Forwarding with SSM Mapping	271
Verifying SSM Mapping Configuration and Operation	272
Monitoring SSM and SSM Mapping	274
Monitoring SSM	274
Monitoring SSM Mapping	275
Configuration Examples for SSM and SSM Mapping	275
SSM with IGMPv3 Example	275
SSM Filtering Example	276
SSM Mapping Example	276
DNS Server Configuration Example	279
Additional References	279
Feature History and Information for SSM	280

PART III
First Hop Protocols 281

CHAPTER 20
Configuring HSRP and VRRP 283

Configuring HSRP and VRRP	283
Information About Configuring HSRP	283
HSRP Overview	283
HSRP Versions	285
Multiple HSRP	286
SSO HSRP	286
HSRP and Switch Stacks	287
How to Configure HSRP	287
Default HSRP Configuration	287
HSRP Configuration Guidelines	287
Enabling HSRP	288
Configuring HSRP Priority	290
Configuring MHSRP	292
Configuring HSRP Authentication and Timers	298
Enabling HSRP Support for ICMP Redirect Messages	300
Configuring HSRP Groups and Clustering	300

Troubleshooting HSRP	300
Verifying HSRP	300
Verifying HSRP Configurations	300
Configuration Examples for Configuring HSRP	301
Enabling HSRP: Example	301
Configuring HSRP Priority: Example	301
Configuring MHSRP: Example	302
Configuring HSRP Authentication and Timer: Example	302
Configuring HSRP Groups and Clustering: Example	302
Information About VRRP	303
Configuring VRRP	303
Additional References for Configuring HSRP	303
Feature Information for HSRP	304

PART IV **IPv6** 305

CHAPTER 21 **Configuring MLD Snooping** 307

Information About Configuring IPv6 MLD Snooping	307
Understanding MLD Snooping	308
MLD Messages	308
MLD Queries	309
Multicast Client Aging Robustness	309
Multicast Router Discovery	309
MLD Reports	310
MLD Done Messages and Immediate-Leave	310
Topology Change Notification Processing	310
MLD Snooping in Switch Stacks	311
How to Configure IPv6 MLD Snooping	311
Default MLD Snooping Configuration	311
MLD Snooping Configuration Guidelines	312
Enabling or Disabling MLD Snooping on the Switch	312
Enabling or Disabling MLD Snooping on a VLAN	313
Configuring a Static Multicast Group	314
Configuring a Multicast Router Port	315

Enabling MLD Immediate Leave	316
Configuring MLD Snooping Queries	317
Disabling MLD Listener Message Suppression	318
Displaying MLD Snooping Information	319
Configuration Examples for Configuring MLD Snooping	320
Configuring a Static Multicast Group: Example	320
Configuring a Multicast Router Port: Example	320
Enabling MLD Immediate Leave: Example	320
Configuring MLD Snooping Queries: Example	321

CHAPTER 22**Configuring IPv6 Unicast Routing 323**

Information About Configuring IPv6 Host Functions	323
Understanding IPv6	323
IPv6 Addresses	324
Supported IPv6 Unicast Routing Features	324
IPv6 and Switch Stacks	328
Default IPv6 Configuration	329
Configuring IPv6 Addressing and Enabling IPv6 Routing	329
Configuring IPv6 ICMP Rate Limiting	331
Configuring Static Routing for IPv6	332
Displaying IPv6	334
Configuration Examples for IPv6 Unicast Routing	335
Configuring IPv6 Addressing and Enabling IPv6 Routing: Example	335
Configuring IPv6 ICMP Rate Limiting: Example	335
Configuring Static Routing for IPv6: Example	336
Displaying IPv6: Example	336

CHAPTER 23**Implementing IPv6 Multicast 337**

Information About Implementing IPv6 Multicast Routing	337
IPv6 Multicast Overview	337
IPv6 Multicast Routing Implementation	338
MLD Access Group	338
Explicit Tracking of Receivers	338
Protocol Independent Multicast	338

PIM-Sparse Mode	338
IPv6 BSR: Configure RP Mapping	339
PIM-Source Specific Multicast	339
Routable Address Hello Option	340
PIM IPv6 Stub Routing	340
Static Mroutes	341
MRIB	341
MFIB	341
Distributed MFIB	342
IPv6 Multicast Process Switching and Fast Switching	342
Implementing IPv6 Multicast	343
Enabling IPv6 Multicast Routing	343
Customizing and Verifying the MLD Protocol	343
Customizing and Verifying MLD on an Interface	343
Implementing MLD Group Limits	345
Configuring Explicit Tracking of Receivers to Track Host Behavior	345
Resetting the MLD Traffic Counters	346
Clearing the MLD Interface Counters	346
Configuring PIM	347
Configuring PIM-SM and Displaying PIM-SM Information for a Group Range	347
Configuring PIM Options	348
Resetting the PIM Traffic Counters	350
Clearing the PIM Topology Table to Reset the MRIB Connection	351
Configuring PIM IPv6 Stub Routing	352
PIM IPv6 Stub Routing Configuration Guidelines	352
Default IPv6 PIM Routing Configuration	353
Enabling IPV6 PIM Stub Routing	353
Monitoring IPv6 PIM Stub Routing	355
Configuring a BSR	355
Configuring a BSR and Verifying BSR Information	355
Sending PIM RP Advertisements to the BSR	356
Configuring BSR for Use Within Scoped Zones	357
Configuring BSR Switches to Announce Scope-to-RP Mappings	358
Configuring SSM Mapping	359

Configuring Static Mroutes	360
Using MFIB in IPv6 Multicast	361
Verifying MFIB Operation in IPv6 Multicast	361
Resetting MFIB Traffic Counters	362

CHAPTER 24**Configuring IPv6 ACL 365**

Information About Configuring IPv6 ACLs	365
Understanding IPv6 ACLs	365
Supported ACL Features	366
IPv6 ACL Limitations	366
Configuring IPv6 ACLs	367
Default IPv6 ACL Configuration	367
Interaction with Other Features and Switches	367
Creating an IPv6 ACL	368
Applying an IPv6 to an Interface	371
Displaying IPv6 ACLs	372
Configuration Examples for IPv6 ACL	373
Example: Creating an IPv6 ACL	373
Example: Applying IPv6 ACLs	373
Example: Displaying IPv6 ACLs	374

PART V**Layer 2/3 375****CHAPTER 25****Configuring Spanning Tree Protocol 377**

Restrictions for STP	377
Information About Spanning Tree Protocol	377
Spanning Tree Protocol	377
Spanning-Tree Topology and BPDUs	378
Bridge ID, Device Priority, and Extended System ID	380
Port Priority Versus Path Cost	381
Spanning-Tree Interface States	381
How a Device or Port Becomes the Root Device or Root Port	384
Spanning Tree and Redundant Connectivity	385
Spanning-Tree Address Management	385

Accelerated Aging to Retain Connectivity	385
Spanning-Tree Modes and Protocols	385
Supported Spanning-Tree Instances	386
Spanning-Tree Interoperability and Backward Compatibility	386
STP and IEEE 802.1Q Trunks	387
VLAN-Bridge Spanning Tree	387
Spanning Tree and Device Stacks	387
Default Spanning-Tree Configuration	388
How to Configure Spanning-Tree Features	389
Changing the Spanning-Tree Mode	389
Disabling Spanning Tree	390
Configuring the Root Device	391
Configuring a Secondary Root Device	392
Configuring Port Priority	393
Configuring Path Cost	395
Configuring the Device Priority of a VLAN	396
Configuring the Hello Time	397
Configuring the Forwarding-Delay Time for a VLAN	398
Configuring the Maximum-Aging Time for a VLAN	399
Configuring the Transmit Hold-Count	400
Monitoring Spanning-Tree Status	401
Additional References for Spanning-Tree Protocol	401
Feature Information for STP	402

CHAPTER 26
Configuring Multiple Spanning-Tree Protocol 403

Prerequisites for MSTP	403
Restrictions for MSTP	403
Information About MSTP	404
MSTP Configuration	404
MSTP Configuration Guidelines	405
Root Switch	405
Multiple Spanning-Tree Regions	406
IST, CIST, and CST	406
Operations Within an MST Region	407

Operations Between MST Regions	407
IEEE 802.1s Terminology	407
Illustration of MST Regions	408
Hop Count	409
Boundary Ports	409
IEEE 802.1s Implementation	410
Port Role Naming Change	410
Interoperation Between Legacy and Standard Devices	411
Detecting Unidirectional Link Failure	411
MSTP and Device Stacks	412
Interoperability with IEEE 802.1D STP	412
RSTP Overview	413
Port Roles and the Active Topology	413
Rapid Convergence	414
Synchronization of Port Roles	415
Bridge Protocol Data Unit Format and Processing	416
Topology Changes	417
Protocol Migration Process	418
Default MSTP Configuration	418
About MST-to-PVST+ Interoperability (PVST+ Simulation)	419
About Detecting Unidirectional Link Failure	420
How to Configure MSTP Features	421
Specifying the MST Region Configuration and Enabling MSTP	421
Configuring the Root Device	423
Configuring a Secondary Root Device	424
Configuring Port Priority	425
Configuring Path Cost	427
Configuring the Device Priority	428
Configuring the Hello Time	430
Configuring the Forwarding-Delay Time	431
Configuring the Maximum-Aging Time	432
Configuring the Maximum-Hop Count	433
Specifying the Link Type to Ensure Rapid Transitions	433
Designating the Neighbor Type	435

Restarting the Protocol Migration Process	436
Configuring PVST+ Simulation	437
Enabling PVST+ Simulation on a Port	438
Examples	439
Examples: PVST+ Simulation	439
Examples: Detecting Unidirectional Link Failure	442
Additional References for MSTP	443
Feature Information for MSTP	444

CHAPTER 27

Configuring Optional Spanning-Tree Features	445
Restriction for Optional Spanning-Tree Features	445
Information About Optional Spanning-Tree Features	445
PortFast	445
BPDU Guard	446
BPDU Filtering	446
UplinkFast	447
Cross-Stack UplinkFast	448
How Cross-Stack UplinkFast Works	449
Events That Cause Fast Convergence	450
BackboneFast	450
EtherChannel Guard	452
Root Guard	453
Loop Guard	454
STP PortFast Port Types	454
Bridge Assurance	455
How to Configure Optional Spanning-Tree Features	457
Enabling PortFast	457
Enabling BPDU Guard	458
Enabling BPDU Filtering	460
Enabling UplinkFast for Use with Redundant Links	461
Disabling UplinkFast	462
Enabling BackboneFast	463
Enabling EtherChannel Guard	464
Enabling Root Guard	465

Enabling Loop Guard	467
Enabling PortFast Port Types	468
Configuring the Default Port State Globally	468
Configuring PortFast Edge on a Specified Interface	469
Configuring a PortFast Network Port on a Specified Interface	470
Enabling Bridge Assurance	471
Examples	472
Examples: Configuring PortFast Edge on a Specified Interface	472
Examples: Configuring a PortFast Network Port on a Specified Interface	473
Example: Configuring Bridge Assurance	474
Monitoring the Spanning-Tree Status	475
Additional References for Optional Spanning Tree Features	475
Feature Information for Optional Spanning-Tree Features	476

CHAPTER 28**Configuring EtherChannels 477**

Restrictions for EtherChannels	477
Information About EtherChannels	477
EtherChannel Overview	477
EtherChannel Modes	478
EtherChannel on Devices	478
EtherChannel Link Failover	479
Channel Groups and Port-Channel Interfaces	479
Port Aggregation Protocol	480
PAgP Modes	481
PAgP Learn Method and Priority	481
PAgP Interaction with Virtual Switches and Dual-Active Detection	482
PAgP Interaction with Other Features	482
Link Aggregation Control Protocol	483
LACP Modes	483
LACP Interaction with Other Features	483
EtherChannel On Mode	484
Load-Balancing and Forwarding Methods	484
MAC Address Forwarding	484
IP Address Forwarding	485

Load-Balancing Advantages	485
EtherChannel Load Deferral Overview	486
EtherChannel and Device Stacks	487
Device Stack and PAgP	487
Switch Stacks and LACP	487
Default EtherChannel Configuration	488
EtherChannel Configuration Guidelines	488
Layer 2 EtherChannel Configuration Guidelines	489
Layer 3 EtherChannel Configuration Guidelines	489
Auto-LAG	490
Auto-LAG Configuration Guidelines	490
How to Configure EtherChannels	491
Configuring Layer 2 EtherChannels	491
Configuring Layer 3 EtherChannels	493
Creating Port-Channel Logical Interfaces	493
Configuring the Physical Interfaces	494
Configuring EtherChannel Load-Balancing	496
Configuring Port Channel Load Deferral	497
Configuring the PAgP Learn Method and Priority	499
Configuring LACP Hot-Standby Ports	500
Configuring the LACP System Priority	501
Configuring the LACP Port Priority	502
Configuring the LACP Port Channel Min-Links Feature	503
Configuring LACP Fast Rate Timer	504
Configuring Auto-LAG Globally	505
Configuring Auto-LAG on a Port Interface	506
Configuring Persistence with Auto-LAG	507
Monitoring EtherChannel, PAgP, and LACP Status	508
Configuration Examples for Configuring EtherChannels	509
Configuring Layer 2 EtherChannels: Examples	509
Configuring Port-Channel Logical Interfaces: Example	510
Configuring EtherChannel Physical Interfaces: Examples	510
Example: Configuring Port Channel Load Deferral	511
Configuring Auto LAG: Examples	511

Configuring LACP Port Channel Min-Links: Examples	512
Example: Configuring LACP Fast Rate Timer	512
Additional References for EtherChannels	513
Feature Information for EtherChannels	514

CHAPTER 29**Configuring Link-State Tracking 515**

Restrictions for Configuring Link-State Tracking	515
Understanding Link-State Tracking	515
How to Configure Link-State Tracking	517
Monitoring Link-State Tracking	518
Configuring Link-State Tracking: Example	519
Additional References for Link-State Tracking	519
Feature Information for Link-State Tracking	520

CHAPTER 30**Configuring Flex Links and the MAC Address-Table Move Update Feature 521**

Restrictions for Configuring Flex Links and MAC Address-Table Move Update	521
Information About Flex Links and MAC Address-Table Move Update	522
Flex Links	522
Flex Links Configuration	522
VLAN Flex Links Load Balancing and Support	523
Multicast Fast Convergence with Flex Links Failover	523
Learning the Other Flex Links Port as the mrouter Port	523
Generating IGMP Reports	523
Leaking IGMP Reports	524
MAC Address-Table Move Update	524
Flex Links VLAN Load Balancing Configuration Guidelines	526
MAC Address-Table Move Update Configuration Guidelines	526
Default Flex Links and MAC Address-Table Move Update Configuration	526
How to Configure Flex Links and the MAC Address-Table Move Update Feature	526
Configuring Flex Links	526
Configuring a Preemption Scheme for a Pair of Flex Links	527
Configuring VLAN Load Balancing on Flex Links	529
Configuring MAC Address-Table Move Update	529
Configuring a Device to Obtain and Process MAC Address-Table Move Update Messages	531

Monitoring Flex Links, Multicast Fast Convergence, and MAC Address-Table Move Update	531
Configuration Examples for Flex Links	532
Configuring Flex Links: Examples	532
Configuring VLAN Load Balancing on Flex Links: Examples	532
Configuring the MAC Address-Table Move Update: Examples	533
Configuring Multicast Fast Convergence with Flex Links Failover: Examples	534
Additional References for Flex Links and MAC Address-Table Move Update	536
Feature Information for Flex Links and MAC Address-Table Move Update	537

CHAPTER 31**Configuring UniDirectional Link Detection 539**

Restrictions for Configuring UDLD	539
Information About UDLD	539
Modes of Operation	539
Normal Mode	540
Aggressive Mode	540
Methods to Detect Unidirectional Links	540
Neighbor Database Maintenance	541
Event-Driven Detection and Echoing	541
UDLD Reset Options	541
Default UDLD Configuration	541
How to Configure UDLD	542
Enabling UDLD Globally	542
Enabling UDLD on an Interface	543
Monitoring and Maintaining UDLD	544
Additional References for UDLD	544
Feature Information for UDLD	545

CHAPTER 32**Configuring Resilient Ethernet Protocol 547**

Overview of Resilient Ethernet Protocol	547
Link Integrity	549
Fast Convergence	550
VLAN Load Balancing	550
Spanning Tree Interaction	551
REP Ports	552

How to Configure Resilient Ethernet Protocol	552
Default REP Configuration	552
REP Configuration Guidelines	552
Configuring REP Administrative VLAN	554
Configuring a REP Interface	555
Setting Manual Preemption for VLAN Load Balancing	559
Configuring SNMP Traps for REP	560
Monitoring Resilient Ethernet Protocol Configurations	560
Configuration Examples for Resilient Ethernet Protocol	562
Example: Configuring the REP Administrative VLAN	562
Example: Configuring a REP Interface	562
<hr/>	
CHAPTER 33	Configuring the PPPoE Intermediate Agent 565
Restrictions for PPPoE Intermediate Agent	565
Information about PPPoE Intermediate Agent	565
How to Configure PPPoE IA	566
Enabling PPPoE IA on a Switch	566
Configuring the Access Node Identifier for PPPoE IA on a Switch	566
Configuring the Identifier String, Option, and Delimiter for PPPoE IA on a Switch	567
Configuring the Generic Error Message for PPPoE IA on a Switch	568
Enabling PPPoE IA on an Interface	568
Configuring the PPPoE IA Trust Setting on an Interface	570
Configuring PPPoE Intermediate Agent Rate Limiting Setting on an Interface	570
Configuring PPPoE IA Vendor-tag Stripping on an Interface	571
Configuring PPPoE Intermediate Agent Circuit-ID and Remote-ID on an Interface	572
Enabling PPPoE IA for a Specific VLAN on an Interface	573
Configuring PPPoE IA Circuit-ID and Remote-ID for a VLAN on an Interface	574
Configuration Examples for PPPoE IA	575
Example: Enabling PPPoE Intermediate Agent on a Switch	575
Example: Configuring the Access Node Identifier for PPPoE IA on a Switch	575
Example: Configuring the Identifier String, Option, and Delimiter for PPPoE IA on a Switch	575
Example: Configuring the Generic Error Message for PPPoE IA on a Switch	576
Example: Enabling PPPoE IA on an Interface	576
Example: Configuring the PPPoE Intermediate Agent Trust Setting on an Interface	576

Example: Configuring PPPoE Intermediate Agent Rate Limiting Setting on an Interface 576

Example: Configuring PPPoE IA Vendor-tag Stripping on an Interface 576

Example: Configuring PPPoE IA Circuit-ID and Remote-ID on an Interface 577

Example: Enabling PPPoE IA for a Specific VLAN on an Interface 577

Example: Configuring PPPoE IA Circuit-ID and Remote-ID for a VLAN on an Interface 577

Displaying Configuration Parameters 577

Clearing Packet Counters 579

Debugging PPPoE Intermediate Agent 579

Troubleshooting Tips 580

Feature Information for Configuring the PPPoE Intermediate Agent 581

PART VI

Network Management 583

CHAPTER 34

Configuring Cisco IOS Configuration Engine 585

Prerequisites for Configuring the Configuration Engine 585

Restrictions for Configuring the Configuration Engine 585

Information About Configuring the Configuration Engine 586

 Cisco Configuration Engine Software 586

 Configuration Service 587

 Event Service 587

 NameSpace Mapper 588

 Cisco Networking Services IDs and Device Hostnames 588

 ConfigID 588

 DeviceID 588

 Hostname and DeviceID 589

 Hostname, DeviceID, and ConfigID 589

 Cisco IOS CNS Agents 589

 Initial Configuration 589

 Incremental (Partial) Configuration 590

 Synchronized Configuration 590

 Automated CNS Configuration 590

How to Configure the Configuration Engine 591

 Enabling the CNS Event Agent 591

 Enabling the Cisco IOS CNS Agent 593

Enabling an Initial Configuration for Cisco IOS CNS Agent	595
Refreshing DeviceIDs	599
Enabling a Partial Configuration for Cisco IOS CNS Agent	601
Monitoring CNS Configurations	603
Additional References	603
Feature History and Information for the Configuration Engine	604

CHAPTER 35**Configuring the Cisco Discovery Protocol 605**

Information About CDP	605
Cisco Discovery Protocol Overview	605
CDP and Stacks	606
Default Cisco Discovery Protocol Configuration	606
How to Configure CDP	606
Configuring Cisco Discovery Protocol Characteristics	606
Disabling Cisco Discovery Protocol	608
Enabling Cisco Discovery Protocol	609
Disabling Cisco Discovery Protocol on an Interface	610
Enabling Cisco Discovery Protocol on an Interface	612
Monitoring and Maintaining Cisco Discovery Protocol	613
Additional References	614
Feature History and Information for Cisco Discovery Protocol	615

CHAPTER 36**Configuring Simple Network Management Protocol 617**

Prerequisites for SNMP	617
Restrictions for SNMP	619
Information About SNMP	619
SNMP Overview	619
SNMP Manager Functions	620
SNMP Agent Functions	620
SNMP Community Strings	621
SNMP MIB Variables Access	621
SNMP Notifications	621
SNMP ifIndex MIB Object Values	622
Default SNMP Configuration	622

SNMP Configuration Guidelines	623
How to Configure SNMP	624
Disabling the SNMP Agent	624
Configuring Community Strings	625
Configuring SNMP Groups and Users	627
Configuring SNMP Notifications	630
Setting the Agent Contact and Location Information	635
Limiting TFTP Servers Used Through SNMP	636
Monitoring SNMP Status	638
SNMP Examples	638
Additional References	639
Feature History and Information for Simple Network Management Protocol	640

CHAPTER 37**Configuring SPAN and RSPAN 641**

Prerequisites for SPAN and RSPAN	641
Restrictions for SPAN and RSPAN	641
Information About SPAN and RSPAN	643
SPAN and RSPAN	643
Local SPAN	643
Remote SPAN	644
SPAN and RSPAN Concepts and Terminology	645
SPAN and RSPAN Interaction with Other Features	650
SPAN and RSPAN and Device Stacks	651
Flow-Based SPAN	651
Default SPAN and RSPAN Configuration	652
Configuration Guidelines	652
SPAN Configuration Guidelines	652
RSPAN Configuration Guidelines	653
FSPAN and FRSPAN Configuration Guidelines	653
How to Configure SPAN and RSPAN	653
Creating a Local SPAN Session	653
Creating a Local SPAN Session and Configuring Incoming Traffic	656
Specifying VLANs to Filter	658
Configuring a VLAN as an RSPAN VLAN	660

Creating an RSPAN Source Session	661
Specifying VLANs to Filter	664
Creating an RSPAN Destination Session	665
Creating an RSPAN Destination Session and Configuring Incoming Traffic	668
Configuring an FSPAN Session	670
Configuring an FRSPAN Session	672
Monitoring SPAN and RSPAN Operations	675
SPAN and RSPAN Configuration Examples	676
Example: Configuring Local SPAN	676
Examples: Creating an RSPAN VLAN	677
Additional References	678
Feature History and Information for SPAN and RSPAN	679

PART VII
Cisco Flexible NetFlow 681

CHAPTER 38
Configuring Flexible NetFlow 683

Prerequisites for Flexible NetFlow	683
Restrictions for Flexible NetFlow	684
Information About Flexible Netflow	686
Flexible NetFlow Overview	686
Flexible NetFlow Components	686
Flow Records	686
Flow Exporters	687
Flow Monitors	688
Flow Samplers	690
Default Settings	690
How to Configure Flexible Netflow	690
Creating a Flow Record	691
Creating a Flow Exporter	694
Creating a Flow Monitor	696
Creating a Sampler	698
Applying a Flow to an Interface	700
Configuring Layer 2 NetFlow	701
Monitoring Flexible NetFlow	703

Configuration Examples for Flexible NetFlow 704
 Example: Configuring a Flow 704
 Additional References for NetFlow 704
 Feature Information for Flexible NetFlow 705

PART VIII

Openflow 707

CHAPTER 39

OpenFlow 709

Finding Feature Information 709
 Prerequisites for OpenFlow 709
 Restrictions for OpenFlow 710
 Information About Open Flow 711
 Overview of OpenFlow 711
 OpenFlow Controller Operation 711
 Cisco OpenFlow Feature Support 712
 Supported Match and Actions and Pipelines 714
 Configuring OpenFlow 717
 Monitoring OpenFlow 721
 Configuration Examples for OpenFlow 721

PART IX

Quality of Service 727

CHAPTER 40

Configuring QoS 729

Prerequisites for QoS 729
 QoS ACL Guidelines 729
 Applying QoS on Interfaces Guidelines 730
 Policing Guidelines 730
 General QoS Guidelines 731
 Restrictions for QoS 731
 Information About QoS 732
 QoS Implementation 732
 Layer 2 Frame Prioritization Bits 733
 Layer 3 Packet Prioritization Bits 733
 End-to-End QoS Solution Using Classification 734

QoS Basic Model	734
Actions at Ingress Port	734
Actions at Egress Port	735
Classification Overview	735
Policing and Marking Overview	740
Mapping Tables Overview	743
Queueing and Scheduling Overview	744
Queueing and Scheduling on Egress Queues	746
Packet Modification	749
Standard QoS Default Configuration	750
Default Egress Queue Configuration	750
Default Mapping Table Configuration	751
DSCP Maps	752
Default CoS-to-DSCP Map	752
Default IP-Precedence-to-DSCP Map	752
Default DSCP-to-CoS Map	753
How to Configure QoS	753
Enabling QoS Globally	753
Enabling VLAN-Based QoS on Physical Ports	754
Configuring Classification Using Port Trust States	755
Configuring the Trust State on Ports Within the QoS Domain	756
Configuring the CoS Value for an Interface	758
Configuring a Trusted Boundary to Ensure Port Security	759
Enabling DSCP Transparency Mode	762
Configuring the DSCP Trust State on a Port Bordering Another QoS Domain	763
Configuring a QoS Policy	766
Classifying Traffic by Using ACLs	766
Classifying Traffic by Using Class Maps	773
Classifying Traffic by Using Class Maps and Filtering IPv6 Traffic	776
Classifying, Policing, and Marking Traffic on Physical Ports by Using Policy Maps	778
Classifying, Policing, and Marking Traffic on SVIs by Using Hierarchical Policy Maps	782
Classifying, Policing, and Marking Traffic by Using Aggregate Policers	791
Configuring DSCP Maps	793
Configuring the CoS-to-DSCP Map	793

Configuring the IP-Precedence-to-DSCP Map	794
Configuring the Policed-DSCP Map	795
Configuring the DSCP-to-CoS Map	796
Configuring the DSCP-to-DSCP-Mutation Map	797
Configuring Ingress Queue Characteristics	799
Configuration Guidelines	800
Mapping DSCP or CoS Values to an Ingress Queue and Setting WTD Thresholds	800
Allocating Buffer Space Between the Ingress Queues	802
Allocating Bandwidth Between the Ingress Queues	803
Configuring Egress Queue Characteristics	804
Configuration Guidelines	805
Allocating Buffer Space to and Setting WTD Thresholds for an Egress Queue Set	805
Mapping DSCP or CoS Values to an Egress Queue and to a Threshold ID	808
Configuring SRR Shaped Weights on Egress Queues	810
Configuring SRR Shared Weights on Egress Queues	812
Configuring the Egress Expedite Queue	813
Limiting the Bandwidth on an Egress Interface	815
Monitoring Standard QoS	816
Configuration Examples for QoS	817
Example: Configuring Port to the DSCP-Trusted State and Modifying the DSCP-to-DSCP-Mutation Map	817
Examples: Classifying Traffic by Using ACLs	817
Examples: Classifying Traffic by Using Class Maps	818
Examples: Classifying, Policing, and Marking Traffic on Physical Ports Using Policy Maps	820
Examples: Classifying, Policing, and Marking Traffic on SVIs by Using Hierarchical Policy Maps	821
Examples: Classifying, Policing, and Marking Traffic by Using Aggregate Policers	823
Examples: Configuring DSCP Maps	824
Examples: Configuring Egress Queue Characteristics	826
Where to Go Next	826
Additional References	827
Feature History and Information for QoS	827

Auto-QoS VoIP Considerations	829
Auto-QoS Enhanced Considerations	830
Restrictions for Auto-QoS	830
Information About Configuring Auto-QoS	830
Auto-QoS Overview	830
Auto-QoS Compact Overview	831
Generated Auto-QoS Configuration	831
VoIP Device Specifics	831
Effects of Auto-QoS on Running Configuration	833
Effects of Auto-QoS Compact on Running Configuration	833
How to Configure Auto-QoS	833
Configuring Auto-QoS	833
Enabling Auto-QoS	833
Enabling Auto-QoS Compact	836
Troubleshooting Auto-QoS	837
Monitoring Auto-QoS	837
Configuration Examples for Auto-QoS	838
Examples: Global Auto-QoS Configuration	838
Examples: Auto-QoS Generated Configuration for VoIP Devices	841
Examples: Auto-QoS Generated Configuration For Enhanced Video, Trust, and Classify Devices	843
auto qos global compact	845
Where to Go Next for Auto-QoS	846
Additional References	846
Feature History and Information for Auto-QoS	847

PART X
Routing 849

CHAPTER 42
Configuring IP Unicast Routing 851

Information About Configuring IP Unicast Routing	851
Information About IP Routing	852
Types of Routing	852
IP Routing and Switch Stacks	853
Classless Routing	854
Address Resolution	855

Proxy ARP	856
ICMP Router Discovery Protocol	856
UDP Broadcast Packets and Protocols	856
Broadcast Packet Handling	857
IP Broadcast Flooding	857
How to Configure IP Routing	858
How to Configure IP Addressing	859
Default IP Addressing Configuration	859
Assigning IP Addresses to Network Interfaces	860
Using Subnet Zero	862
Disabling Classless Routing	863
Configuring Address Resolution Methods	864
Defining a Static ARP Cache	864
Setting ARP Encapsulation	865
Enabling Proxy ARP	866
Routing Assistance When IP Routing is Disabled	867
Proxy ARP	868
Default Gateway	868
ICMP Router Discovery Protocol (IRDP)	869
Configuring Broadcast Packet Handling	870
Enabling Directed Broadcast-to-Physical Broadcast Translation	871
Forwarding UDP Broadcast Packets and Protocols	872
Establishing an IP Broadcast Address	874
Flooding IP Broadcasts	875
Monitoring and Maintaining IP Addressing	876
How to Configure IP Unicast Routing	877
Enabling IP Unicast Routing	877
Example of Enabling IP Routing	878
What to Do Next	878
Information About RIP	878
Summary Addresses and Split Horizon	879
How to Configure RIP	879
Default RIP Configuration	879
Configuring Basic RIP Parameters	880

Configuring RIP Authentication	882
Configuring Summary Addresses and Split Horizon	883
Configuring Split Horizon	885
Configuration Example for Summary Addresses and Split Horizon	886
Information About OSPF	886
OSPF for Routed Access	887
OSPF Area Parameters	887
Other OSPF Parameters	887
LSA Group Pacing	888
Loopback Interfaces	888
How to Configure OSPF	889
Default OSPF Configuration	889
Configuring Basic OSPF Parameters	890
Configuring OSPF Interfaces	891
Configuring OSPF Area Parameters	893
Configuring Other OSPF Parameters	895
Changing LSA Group Pacing	897
Configuring a Loopback Interface	898
Monitoring OSPF	899
Configuration Examples for OSPF	900
Example: Configuring Basic OSPF Parameters	900
Information About EIGRP	900
EIGRP Stub Routing	900
Configuring Unicast Reverse Path Forwarding	901
Protocol-Independent Features	901
Distributed Cisco Express Forwarding	902
Information About Cisco Express Forwarding	902
How to Configure Cisco Express Forwarding	902
Number of Equal-Cost Routing Paths	904
Information About Equal-Cost Routing Paths	904
How to Configure Equal-Cost Routing Paths	904
Static Unicast Routes	905
Information About Static Unicast Routes	905
Configuring Static Unicast Routes	906

Default Routes and Networks	907
Information About Default Routes and Networks	907
How to Configure Default Routes and Networks	908
Route Maps to Redistribute Routing Information	908
Information About Route Maps	908
How to Configure a Route Map	909
How to Control Route Distribution	911
Policy-Based Routing	913
Information About Policy-Based Routing	913
How to Configure PBR	914
Filtering Routing Information	916
Setting Passive Interfaces	916
Controlling Advertising and Processing in Routing Updates	918
Filtering Sources of Routing Information	919
Managing Authentication Keys	920
Prerequisites	920
How to Configure Authentication Keys	920
Monitoring and Maintaining the IP Network	921
<hr/>	
CHAPTER 43	Configuring Bidirectional Forwarding Detection 923
	Prerequisites for Bidirectional Forwarding Detection 923
	Restrictions for Bidirectional Forwarding Detection 923
	Information About Bidirectional Forwarding Detection 924
	BFD Operation 924
	BFD Version Interoperability 924
	BFD Session Limits 924
	BFD Support for Nonbroadcast Media Interfaces 924
	BFD Support for Nonstop Forwarding with Stateful Switchover 924
	BFD Support for Stateful Switchover 925
	Stateful BFD on the Standby RP 925
	BFD Support for Static Routing 926
	Configuring BFD Echo Mode 926
	Prerequisites 926
	Restrictions 926

How to Configure Bidirectional Forwarding Detection	927
Configuring BFD Session Parameters on the Interface	927
Configuring BFD Support for Static Routing	928
Configuring the BFD Slow Timer	930
Disabling BFD Echo Mode Without Asymmetry	931
Monitoring and Troubleshooting BFD	931
Configuration Examples for Bidirectional Forwarding Detection	932
Example: Configuring BFD Session Parameters on the Interface	932
Example: Configuring BFD Support for Static Routing	932
Example: Configuring BFD Slow Timer	933
Additional References for Bidirectional Forwarding Detection	933
Feature Information for Bidirectional Forwarding Detection	934

CHAPTER 44
Bidirectional Forwarding Detection 935

Prerequisites for Bidirectional Forwarding Detection	935
Restrictions for Bidirectional Forwarding Detection	935
Information About Bidirectional Forwarding Detection	936
BFD Operation	936
BFD Version Interoperability	936
BFD Session Limits	936
BFD Support for Nonbroadcast Media Interfaces	936
BFD Support for Nonstop Forwarding with Stateful Switchover	937
BFD Support for Stateful Switchover	937
Stateful BFD on the Standby RP	937
BFD Support for Static Routing	938
Configuring BFD Echo Mode	938
Prerequisites	938
Restrictions	939
How to Configure Bidirectional Forwarding Detection	939
Configuring BFD Session Parameters on the Interface	939
Configuring BFD Support for Static Routing	940
Configuring the BFD Slow Timer	942
Disabling BFD Echo Mode Without Asymmetry	943
Monitoring and Troubleshooting BFD	944

Configuration Examples for Bidirectional Forwarding Detection	944
Example: Configuring BFD Session Parameters on the Interface	944
Example: Configuring BFD Support for Static Routing	944
Example: Configuring BFD Slow Timer	945
Additional References for Bidirectional Forwarding Detection	946
Feature Information for Bidirectional Forwarding Detection	946

CHAPTER 45**Configuring IPv6 First Hop Security 947**

Finding Feature Information	947
Prerequisites for First Hop Security in IPv6	947
Restrictions for First Hop Security in IPv6	948
Information about First Hop Security in IPv6	948
How to Configure an IPv6 Snooping Policy	951
How to Attach an IPv6 Snooping Policy to an Interface	953
How to Attach an IPv6 Snooping Policy to a Layer 2 EtherChannel Interface	954
How to Configure the IPv6 Binding Table Content	955
How to Configure an IPv6 Neighbor Discovery Inspection Policy	956
How to Attach an IPv6 Neighbor Discovery Inspection Policy to an Interface	958
How to Attach an IPv6 Neighbor Discovery Inspection Policy to a Layer 2 EtherChannel Interface	959
How to Attach an IPv6 Neighbor Discovery Multicast Suppress Policy on a Device	960
How to Attach an IPv6 Neighbor Discovery Multicast Suppress Policy on an Interface	961
How to Attach an IPv6 Neighbor Discovery Multicast Suppress Policy to a Layer 2 EtherChannel Interface	962
How to Configure an IPv6 Router Advertisement Guard Policy	963
How to Attach an IPv6 Router Advertisement Guard Policy to an Interface	965
How to Attach an IPv6 Router Advertisement Guard Policy to a Layer 2 EtherChannel Interface	966
How to Configure an IPv6 DHCP Guard Policy	968
How to Attach an IPv6 DHCP Guard Policy to an Interface or a VLAN on an Interface	970
How to Attach an IPv6 DHCP Guard Policy to a Layer 2 EtherChannel Interface	971
How to Configure IPv6 Source Guard	972
How to Attach an IPv6 Source Guard Policy to an Interface	973
How to attach an IPv6 Source Guard Policy to a Layer 2 EtherChannel Interface	974
How to Configure IPv6 Prefix Guard	975

How to Attach an IPv6 Prefix Guard Policy to an Interface	976
How to attach an IPv6 Prefix Guard Policy to a Layer 2 EtherChannel Interface	977
Configuration Examples for IPv6 First Hop Security	978
Examples: How to attach an IPv6 Source Guard Policy to a Layer 2 EtherChannel Interface	978
Examples: How to attach an IPv6 Prefix Guard Policy to a Layer 2 EtherChannel Interface	978
Additional References	979

CHAPTER 46**Routing Information Protocol 981**

Prerequisites for RIP	981
Restrictions for RIP	981
Information About Routing Information Protocol	981
RIP Overview	981
RIP Routing Updates	982
Authentication in RIP	982
RIP Routing Metric	983
RIP Versions	983
Exchange of Routing Information	983
Neighbor Router Authentication	984
How to Configure Routing Information Protocol	985
Enabling RIP and Configuring RIP Parameters	985
Specifying a RIP Version and Enabling Authentication	986
Configuration Examples for Routing Information Protocol	988
Example: Enabling RIP and Configuring RIP Parameters	988
Example: Specifying a RIP Version and Enabling Authentication	988
Additional References for RIP	989
Feature Information for RIP	989

PART XI**Security 991****CHAPTER 47****Security Features Overview 993**

Security Features Overview	993
----------------------------	-----

CHAPTER 48**Preventing Unauthorized Access 997**

Preventing Unauthorized Access	997
--------------------------------	-----

CHAPTER 49**Controlling Switch Access with Passwords and Privilege Levels 999**

Restrictions for Controlling Switch Access with Passwords and Privileges 999

Information About Passwords and Privilege Levels 999

Default Password and Privilege Level Configuration 999

Additional Password Security 1000

Password Recovery 1000

Terminal Line Telnet Configuration 1001

Username and Password Pairs 1001

Privilege Levels 1001

How to Control Switch Access with Passwords and Privilege Levels 1002

Setting or Changing a Static Enable Password 1002

Protecting Enable and Enable Secret Passwords with Encryption 1003

Disabling Password Recovery 1005

Setting a Telnet Password for a Terminal Line 1006

Configuring Username and Password Pairs 1007

Setting the Privilege Level for a Command 1009

Changing the Default Privilege Level for Lines 1011

Logging into and Exiting a Privilege Level 1012

Monitoring Switch Access 1012

Configuration Examples for Setting Passwords and Privilege Levels 1013

Example: Setting or Changing a Static Enable Password 1013

Example: Protecting Enable and Enable Secret Passwords with Encryption 1013

Example: Setting a Telnet Password for a Terminal Line 1013

Example: Setting the Privilege Level for a Command 1013

Additional References 1013

CHAPTER 50**Configuring TACACS+ 1015**

Finding Feature Information 1015

Prerequisites for TACACS+ 1015

Restrictions for TACACS+ 1016

Information About TACACS+ 1017

TACACS+ and Switch Access 1017

TACACS+ Overview 1017

TACACS+ Operation	1018
Method List	1019
TACACS AV Pairs	1019
TACACS Authentication and Authorization AV Pairs	1019
TACACS Accounting AV Pairs	1027
Configuring AAA Server Group Selection Based on DNIS	1038
TACACS+ Configuration Options	1040
TACACS+ Login Authentication	1040
TACACS+ Authorization for Privileged EXEC Access and Network Services	1040
TACACS+ Authentication	1040
TACACS+ Authorization	1041
TACACS+ Accounting	1041
Default TACACS+ Configuration	1041
Per VRF for TACACS Servers	1041
How to Configure TACACS+	1041
Identifying the TACACS+ Server Host and Setting the Authentication Key	1041
Configuring TACACS+ Login Authentication	1043
Configuring TACACS+ Authorization for Privileged EXEC Access and Network Services	1045
Starting TACACS+ Accounting	1047
Establishing a Session with a Router if the AAA Server is Unreachable	1048
Establishing a Session with a Router if the AAA Server is Unreachable	1049
Configuring Per VRF on a TACACS Server	1049
Verifying Per VRF for TACACS Servers	1051
Monitoring TACACS+	1052
Configuration Examples for TACACS+	1052
Example: TACACS Authorization	1052
Example: TACACS Accounting	1053
Example: TACACS Authentication	1053
Example: Configuring Per VRF for TACACS Servers	1055
Additional References for TACACS+	1056
Feature Information for TACACS+	1056
<hr/>	
CHAPTER 51	Configuring RADIUS 1059
	Prerequisites for Configuring RADIUS 1059

Restrictions for Configuring RADIUS	1060
Information about RADIUS	1060
RADIUS and Switch Access	1060
RADIUS Overview	1060
RADIUS Operation	1061
Default RADIUS Configuration	1062
RADIUS Server Host	1062
RADIUS Login Authentication	1062
AAA Server Groups	1063
AAA Authorization	1063
RADIUS Accounting	1063
Vendor-Specific RADIUS Attributes	1063
RADIUS Disconnect-Cause Attribute Values	1075
RADIUS Progress Codes	1079
Vendor-Proprietary RADIUS Server Communication	1079
Enhanced Test Command	1080
How to Configure RADIUS	1080
Identifying the RADIUS Server Host	1080
Configuring Settings for All RADIUS Servers	1082
Configuring RADIUS Login Authentication	1083
Defining AAA Server Groups	1086
Configuring RADIUS Authorization for User Privileged Access and Network Services	1088
Starting RADIUS Accounting	1089
Verifying Attribute 196	1090
Configuring the Device to Use Vendor-Specific RADIUS Attributes	1091
Configuring the Device for Vendor-Proprietary RADIUS Server Communication	1092
Configuring a User Profile and Associating it with the RADIUS Record	1094
Verifying the Enhanced Test Command Configuration	1095
Configuration Examples for RADIUS	1095
Examples: Identifying the RADIUS Server Host	1095
Example: Using Two Different RADIUS Group Servers	1095
Examples: AAA Server Groups	1096
Troubleshooting Tips for RADIUS Progress Codes	1096
Examples: Configuring the Switch to Use Vendor-Specific RADIUS Attributes	1097

Example: Configuring the Switch for Vendor-Proprietary RADIUS Server Communication	1097
Example: User Profile Associated With the test aaa group Command	1097
Additional References for RADIUS	1098
Feature Information for RADIUS	1099

CHAPTER 52**RADIUS Server Load Balancing 1101**

Finding Feature Information	1101
Prerequisites for RADIUS Server Load Balancing	1101
Restrictions for RADIUS Server Load Balancing	1102
Information About RADIUS Server Load Balancing	1102
RADIUS Server Load Balancing Overview	1102
Transaction Load Balancing Across RADIUS Server Groups	1102
RADIUS Server Status and Automated Testing	1103
How to Configure RADIUS Server Load Balancing	1104
Enabling Load Balancing for a Named RADIUS Server Group	1104
Enabling Load Balancing for a Global RADIUS Server Group	1105
Troubleshooting RADIUS Server Load Balancing	1106
Configuration Examples for RADIUS Server Load Balancing	1108
Example: Enabling Load Balancing for a Named RADIUS Server Group	1108
Example: Enabling Load Balancing for a Global RADIUS Server Group	1110
Example: Monitoring Idle Timer	1112
Example: Configuring the Preferred Server with the Same Authentication and Authorization Server	1113
Example: Configuring the Preferred Server with Different Authentication and Authorization Servers	1113
Example: Configuring the Preferred Server with Overlapping Authentication and Authorization Servers	1113
Example: Configuring the Preferred Server with Authentication Servers As a Subset of Authorization Servers	1114
Example: Configuring the Preferred Server with Authentication Servers As a Superset of Authorization Servers	1114
Additional References for RADIUS Server Load Balancing	1115
Feature Information for RADIUS Server Load Balancing	1115

CHAPTER 53**RADIUS Change of Authorization Support 1117**

Information About RADIUS Change-of-Authorization	1117
RADIUS Change of Authorization	1117
Change-of-Authorization Requests	1119
RFC 5176 Compliance	1119
Preconditions	1120
CoA Request Response Code	1120
Session Identification	1120
Session Identification	1121
CoA ACK Response Code	1122
CoA NAK Response Code	1122
Session Reauthentication	1122
Session Reauthentication in a Switch Stack	1122
Session Termination	1123
CoA Activate Service Command	1123
CoA Deactivate Service Command	1124
CoA Request: Disable Host Port	1125
CoA Request: Bounce-Port	1125
CoA Session Query Command	1126
CoA Session Reauthenticate Command	1126
CoA Session Terminate Command	1127
Stacking Guidelines for Session Termination	1127
Stacking Guidelines for CoA-Request Bounce-Port	1127
Stacking Guidelines for CoA-Request Disable-Port	1128
How to Configure RADIUS Change-of-Authorization	1128
Configuring CoA on the Device	1128
Monitoring and Troubleshooting CoA Functionality	1130
Additional References for RADIUS Change-of-Authorization	1131
Feature Information for RADIUS Change-of-Authorization Support	1131

CHAPTER 54
Local AAA Server 1133

Prerequisites for Local AAA Server	1133
Information About Local AAA Server	1133
Local Authorization Attributes Overview	1133
Local AAA Attribute Support	1133

AAA Attribute Lists	1134
Converting from RADIUS Format to Cisco IOS AAA Format	1134
Validation of Attributes	1134
How to Configure a Local AAA Server	1135
Defining a AAA Attribute List	1135
Defining a Subscriber Profile	1136
Monitoring and Troubleshooting a Local AAA Server	1138
Configuration Examples for Local AAA Server	1139
Example: Local AAA Server	1139
Example: Mapping from the RADIUS Version of a Particular Attribute to the Cisco IOS AAA Version	1140
Additional References for Local AAA Server	1141
Feature Information for Local AAA Server	1141

CHAPTER 55**Configuring Kerberos 1143**

Prerequisites for Controlling Switch Access with Kerberos	1143
Restrictions for Controlling Switch Access with Kerberos	1143
Information about Kerberos	1144
Kerberos and Switch Access	1144
Kerberos Overview	1144
Kerberos Operation	1146
Authenticating to a Boundary Switch	1146
Obtaining a TGT from a KDC	1147
Authenticating to Network Services	1147
How to Configure Kerberos	1147
Monitoring the Kerberos Configuration	1147
Additional References	1147
Feature Information for Kerberos	1148

CHAPTER 56**Configuring Accounting 1149**

Prerequisites for Configuring Accounting	1149
Restrictions for Configuring Accounting	1149
Information About Configuring Accounting	1150
Named Method Lists for Accounting	1150

Method Lists and Server Groups	1151
AAA Accounting Methods	1151
Accounting Record Types	1152
AAA Accounting Methods	1152
AAA Accounting Types	1152
Network Accounting	1152
EXEC Accounting	1155
Command Accounting	1156
Connection Accounting	1157
System Accounting	1158
Resource Accounting	1159
VRRS Accounting	1161
AAA Accounting Enhancements	1162
AAA Broadcast Accounting	1162
AAA Session MIB	1162
Accounting Attribute-Value Pairs	1163
How to Configure Accounting	1163
Configuring AAA Accounting Using Named Method Lists	1163
Configuring RADIUS System Accounting	1164
Suppressing Generation of Accounting Records for Null Username Sessions	1166
Generating Interim Accounting Records	1166
Generating Accounting Records for Failed Login or Session	1167
Specifying Accounting NETWORK-Stop Records Before EXEC-Stop Records	1167
Configuring AAA Resource Failure Stop Accounting	1167
Configuring AAA Resource Accounting for Start-Stop Records	1168
Configuring AAA Broadcast Accounting	1168
Configuring Per-DNIS AAA Broadcast Accounting	1168
Configuring AAA Session MIB	1169
Configuring VRRS Accounting	1169
Establishing a Session with a Device if the AAA Server is Unreachable	1171
Monitoring Accounting	1171
Troubleshooting Accounting	1172
Configuration Examples for Accounting	1172
Example Configuring Named Method List	1172

Example Configuring AAA Resource Accounting	1174
Example Configuring AAA Broadcast Accounting	1174
Example Configuring Per-DNIS AAA Broadcast Accounting	1175
Example AAA Session MIB	1175
Example Configuring VRRS Accounting	1175
Additional References for Configuring Accounting	1176
Feature Information for Configuring Accounting	1176

CHAPTER 57**Configuring Local Authentication and Authorization 1179**

How to Configure Local Authentication and Authorization	1179
Configuring the Switch for Local Authentication and Authorization	1179
Monitoring Local Authentication and Authorization	1181
Additional References	1182
Feature Information for Local Authentication and Authorization	1182

CHAPTER 58**Certification Authority Interoperability 1183**

Prerequisites For Certification Authority	1183
Restrictions for Certification Authority	1183
Information About Certification Authority	1183
CA Supported Standards	1183
Purpose of CAs	1184
Implementing IPsec Without CAs	1185
Implementing IPsec With CAs	1185
Implementing IPsec with Multiple Root CAs	1185
How CA Certificates Are Used by IPsec Devices	1186
Registration Authorities	1186
How to Configure Certification Authority	1186
Managing NVRAM Memory Usage	1186
Configuring the Device Host Name and IP Domain Name	1188
Generating an RSA Key Pair	1188
Declaring a Certification Authority	1189
Configuring a Root CA (Trusted Root)	1191
Authenticating the CA	1192
Requesting Signed Certificates	1193

Monitoring and Maintaining Certification Authority	1194
Requesting a Certificate Revocation List	1194
Querying a Certification Revocation List	1195
Deleting RSA Keys from a Device	1195
Deleting Public Keys for a Peer	1196
Deleting Certificates from the Configuration	1197
Viewing Keys and Certificates	1198

CHAPTER 59**MAC Authentication Bypass 1201**

Prerequisites for Configuring MAC Authentication Bypass	1201
Information About MAC Authentication Bypass	1202
Overview of the Cisco IOS Auth Manager	1202
Overview of the Configurable MAB Username and Password	1202
How to Configure MAC Authentication Bypass	1203
Enabling MAC Authentication Bypass	1203
Enabling Reauthentication on a Port	1204
Specifying the Security Violation Mode	1206
Enabling Configurable MAB Username and Password	1208
Configuration Examples for MAC Authentication Bypass	1208
Example: MAC Authentication Bypass Configuration	1208
Example: Enabling Configurable MAB Username and Password	1209
Additional References for MAC Authentication Bypass	1209
Feature Information for MAC Authentication Bypass	1210

CHAPTER 60**Password Strength and Management for Common Criteria 1211**

Restrictions for Password Strength and Management for Common Criteria	1211
Information About Password Strength and Management for Common Criteria	1211
Password Composition Policy	1211
Password Length Policy	1212
Password Lifetime Policy	1212
Password Expiry Policy	1212
Password Change Policy	1212
User Reauthentication Policy	1213
Support for Framed (Noninteractive) Session	1213

How to Configure Password Strength and Management for Common Criteria	1213
Configuring the Password Security Policy	1213
Verifying the Common Criteria Policy	1215
Configuration Examples for Password Strength and Management for Common Criteria	1216
Example: Password Strength and Management for Common Criteria	1216
Additional References for Password Strength and Management for Common Criteria	1217
Feature Information for Password Strength and Management for Common Criteria	1217

CHAPTER 61**AAA-SERVER-MIB Set Operation 1219**

Prerequisites for AAA-SERVER-MIB Set Operation	1219
Restrictions for AAA-SERVER-MIB Set Operation	1219
Information About AAA-SERVER-MIB Set Operation	1219
CISCO-AAA-SERVER-MIB	1219
CISCO-AAA-SERVER-MIB Set Operation	1220
How to Configure AAA-SERVER-MIB Set Operation	1220
Configuring AAA-SERVER-MIB Set Operations	1220
Verifying SNMP Values	1220
Configuration Examples for AAA-SERVER-MIB Set Operation	1221
RADIUS Server Configuration and Server Statistics Example	1221
Additional References for AAA-SERVER-MIB Set Operation	1223
Feature Information for AAA-SERVER-MIB Set Operation	1223

CHAPTER 62**Configuring Secure Shell 1225**

Prerequisites for Configuring Secure Shell	1225
Restrictions for Configuring Secure Shell	1226
Information About Configuring Secure Shell	1226
SSH and Switch Access	1226
SSH Servers, Integrated Clients, and Supported Versions	1226
RSA Authentication Support	1227
SSL Configuration Guidelines	1227
Secure Copy Protocol Overview	1227
Secure Copy Protocol	1228
How Secure Copy Works	1228
Reverse Telnet	1228

Reverse SSH	1228
How to Configure Secure Shell	1229
Setting Up the Device to Run SSH	1229
Configuring the SSH Server	1230
Invoking an SSH Client	1232
Troubleshooting Tips	1233
Configuring Reverse SSH for Console Access	1233
Configuring Reverse SSH for Modem Access	1235
Troubleshooting Reverse SSH on the Client	1237
Troubleshooting Reverse SSH on the Server	1237
Monitoring the SSH Configuration and Status	1238
Configuring Secure Copy	1238
Configuration Examples for Secure Shell	1240
Example: Secure Copy Configuration Using Local Authentication	1240
Example: SCP Server-Side Configuration Using Network-Based Authentication	1240
Example Reverse SSH Console Access	1241
Example Reverse SSH Modem Access	1241
Example: Monitoring the SSH Configuration and Status	1241
Additional References for Secure Shell	1242
Feature Information for Configuring Secure Shell	1242
<hr/>	
CHAPTER 63	Secure Shell Version 2 Support 1245
Information About Secure Shell Version 2 Support	1245
Secure Shell Version 2	1245
Secure Shell Version 2 Enhancements	1246
Secure Shell Version 2 Enhancements for RSA Keys	1246
SNMP Trap Generation	1247
SSH Keyboard Interactive Authentication	1248
How to Configure Secure Shell Version 2 Support	1248
Configuring a Device for SSH Version 2 Using a Hostname and Domain Name	1248
Configuring a Device for SSH Version 2 Using RSA Key Pairs	1249
Configuring the Cisco SSH Server to Perform RSA-Based User Authentication	1251
Configuring the Cisco IOS SSH Client to Perform RSA-Based Server Authentication	1252
Starting an Encrypted Session with a Remote Device	1255

Enabling Secure Copy Protocol on the SSH Server	1255
Verifying the Status of the Secure Shell Connection	1257
Verifying the Secure Shell Status	1258
Monitoring and Maintaining Secure Shell Version 2	1260
Configuration Examples for Secure Shell Version 2 Support	1263
Example: Configuring Secure Shell Version 2	1263
Example: Starting an Encrypted Session with a Remote Device	1263
Example: Configuring Server-Side SCP	1263
Example: Setting an SNMP Trap	1263
Examples: SSH Keyboard Interactive Authentication	1264
Example: Enabling Client-Side Debugs	1264
Example: Enabling ChPass with a Blank Password Change	1265
Example: Enabling ChPass and Changing the Password on First Login	1265
Example: Enabling ChPass and Expiring the Password After Three Logins	1265
Example: SNMP Debugging	1266
Examples: SSH Debugging Enhancements	1266
Additional References for Secure Shell Version 2 Support	1268
Feature Information for Secure Shell Version 2 Support	1268

CHAPTER 64
X.509v3 Certificates for SSH Authentication 1271

Prerequisites for X.509v3 Certificates for SSH Authentication	1271
Restrictions for X.509v3 Certificates for SSH Authentication	1271
Information About X.509v3 Certificates for SSH Authentication	1272
X.509v3 Certificates for SSH Authentication Overview	1272
Server and User Authentication Using X.509v3	1272
OCSP Response Stapling	1272
How to Configure X.509v3 Certificates for SSH Authentication	1273
Configuring Digital Certificates for Server Authentication	1273
Configuring Digital Certificates for User Authentication	1274
Verifying the Server and User Authentication Using Digital Certificates	1276
Configuration Examples for X.509v3 Certificates for SSH Authentication	1280
Example: Configuring Digital Certificates for Server Authentication	1280
Example: Configuring Digital Certificate for User Authentication	1280
Additional References for X.509v3 Certificates for SSH Authentication	1281

Feature Information for X.509v3 Certificates for SSH Authentication 1281

CHAPTER 65**Configuring Secure Socket Layer HTTP 1283**

- Information About Secure Socket Layer HTTP 1283
 - Secure HTTP Servers and Clients Overview 1283
 - Certificate Authority Trustpoints 1284
 - CipherSuites 1285
 - Default SSL Configuration 1286
 - SSL Configuration Guidelines 1286
 - How to Configure Secure Socket Layer HTTP 1286
 - Configuring the Secure HTTP Server 1286
 - Configuring the Secure HTTP Client 1290
 - Configuring a CA Trustpoint 1291
- Monitoring Secure HTTP Server and Client Status 1293
- Configuration Examples for Secure Socket Layer HTTP 1294
 - Example: Configuring Secure Socket Layer HTTP 1294
- Additional References for Secure Socket Layer HTTP 1295
- Feature Information for Secure Socket Layer HTTP 1295
- Glossary 1295

CHAPTER 66**Access Control List Overview 1297**

- Information About Access Control Lists 1297
 - Definition of an Access List 1297
 - Functions of an Access Control List 1298
 - Purpose of IP Access Lists 1298
 - Reasons to Configure ACLs 1298
 - Software Processing of an Access List 1299
 - Access List Rules 1299
 - Helpful Hints for Creating IP Access Lists 1300
 - IP Packet Fields You Can Filter to Control Access 1301
 - Source and Destination Addresses 1301
 - Wildcard Mask for Addresses in an Access List 1301
 - Access List Sequence Numbers 1302
 - ACL Supported Types 1302

Supported ACLs	1303
ACL Precedence	1303
Port ACLs	1303
Router ACLs	1304
Access Control Entries	1305
ACEs and Fragmented and Unfragmented Traffic	1305
ACEs and Fragmented and Unfragmented Traffic Examples	1305

CHAPTER 67
Configuring IPv4 Access Control Lists 1307

Prerequisites for Configuring IPv4 Access Control Lists	1307
Restrictions for Configuring IPv4 Access Control Lists	1307
Information About Configuring IPv4 Access Control Lists	1308
ACL Overview	1308
Standard and Extended IPv4 ACLs	1309
IPv4 ACL Switch Unsupported Features	1309
Access List Numbers	1309
Numbered Standard IPv4 ACLs	1310
Numbered Extended IPv4 ACLs	1310
Named IPv4 ACLs	1311
Benefits of IP Access List Entry Sequence Numbering	1312
Sequence Numbering Behavior	1312
Including comments in ACLs	1313
Hardware and Software Treatment of IP ACLs	1313
Time Ranges for ACLs	1313
IPv4 ACL Interface Considerations	1314
Apply an Access Control List to an Interface	1314
ACL Logging	1315
How to Configure ACLs	1316
Configuring IPv4 ACLs	1316
Creating a Numbered Standard ACL	1316
Creating a Numbered Extended ACL	1318
Creating Named Standard ACLs	1321
Creating Extended Named ACLs	1323
Configuring an Access Control Entry with Noncontiguous Ports	1324

Consolidating Access List Entries with Noncontiguous Ports into One Access List Entry	1326
Sequencing Access-List Entries and Revising the Access List	1328
Configuring Commented IP ACL Entries	1331
Configuring Time Ranges for ACLs	1332
Applying an IPv4 ACL to a Terminal Line	1334
Applying an IPv4 ACL to an Interface (CLI)	1335
Monitoring IPv4 ACLs	1336
Configuration Examples for ACLs	1337
ACLs in a Small Networked Office	1337
Example: Numbered ACLs	1338
Examples: Extended ACLs	1338
Examples: Named ACLs	1339
Example: Configuring an Access Control Entry with Noncontiguous Ports	1340
Example: Consolidating Access List Entries with Noncontiguous Ports into One Access List Entry	1340
Example Resequencing Entries in an Access List	1341
Example Adding an Entry with a Sequence Number	1341
Example Adding an Entry with No Sequence Number	1342
Examples: Configuring Commented IP ACL Entries	1342
Examples: Using Time Ranges with ACLs	1343
Examples: Time Range Applied to an IP ACL	1344
Examples: ACL Logging	1344
Examples: Troubleshooting ACLs	1345
Additional References	1346
Feature Information for IPv4 Access Control Lists	1347

CHAPTER 68**IPv6 Access Control Lists 1349**

Prerequisites for IPv6 ACLs	1349
Restrictions for IPv6 ACLs	1349
Information About Configuring IPv6 ACLs	1350
ACL Overview	1350
IPv6 ACLs Overview	1351
Understanding IPv6 ACLs	1351
Interactions with Other Features and Switches	1352

Default Configuration for IPv6 ACLs	1352
Supported ACL Features	1352
IPv6 Port-Based Access Control List Support	1353
ACLs and Traffic Forwarding	1353
How to Configure IPv6 ACLs	1353
Configuring IPv6 ACLs	1353
Attaching an IPv6 ACL to an Interface	1357
Monitoring IPv6 ACLs	1358
Configuring PACL Mode and Applying IPv6 PACL on an Interface	1359
Configuring IPv6 ACL Extensions for Hop by Hop Filtering	1360
Configuration Examples for IPv6 ACLs	1362
Example: Configuring IPv6 ACLs	1362
Example: Applying IPv6 ACLs	1362
Example: Configuring PACL Mode and Applying IPv6 PACL on an Interface	1362
Example: IPv6 ACL Extensions for Hop by Hop Filtering	1362
Additional References	1363
Feature Information for IPv6 Access Control Lists	1364

CHAPTER 69

ACL Support for Filtering IP Options	1367
Prerequisites for ACL Support for Filtering IP Options	1367
Information About ACL Support for Filtering IP Options	1367
IP Options	1367
Benefits of Filtering IP Options	1368
Benefits of Filtering on TCP Flags	1368
TCP Flags	1368
How to Configure ACL Support for Filtering IP Options	1369
Filtering Packets That Contain IP Options	1369
Filtering Packets That Contain TCP Flags	1371
Configuration Examples for ACL Support for Filtering IP Options	1373
Example: Filtering Packets That Contain IP Options	1373
Example: Filtering Packets That Contain TCP Flags	1373
Additional References for ACL Support for Filtering IP Options	1374
Feature Information for Creating an IP Access List to Filter	1374

CHAPTER 70

VLAN Access Control Lists 1377

- Information About VLAN Access Control Lists 1377
 - VLAN Maps 1377
 - VLAN Map Configuration Guidelines 1378
 - VLAN Maps with Router ACLs 1378
 - VLAN Maps and Router ACL Configuration Guidelines 1379
 - VACL Logging 1379
- How to Configure VLAN Access Control Lists 1380
 - Creating Named MAC Extended ACLs 1380
 - Applying a MAC ACL to a Layer 2 Interface 1382
 - Configuring VLAN Maps 1383
 - Creating a VLAN Map 1385
 - Applying a VLAN Map to a VLAN 1386
 - Configuring VACL Logging 1387
- Configuration Examples for ACLs and VLAN Maps 1389
 - Example: Creating an ACL and a VLAN Map to Deny a Packet 1389
 - Example: Creating an ACL and a VLAN Map to Permit a Packet 1389
 - Example: Default Action of Dropping IP Packets and Forwarding MAC Packets 1389
 - Example: Default Action of Dropping MAC Packets and Forwarding IP Packets 1390
 - Example: Default Action of Dropping All Packets 1390
- Configuration Examples for Using VLAN Maps in Your Network 1391
 - Example: Wiring Closet Configuration 1391
 - Example: Restricting Access to a Server on Another VLAN 1392
 - Example: Denying Access to a Server on Another VLAN 1392
- Configuration Examples of Router ACLs and VLAN Maps Applied to VLANs 1393
 - Example: ACLs and Switched Packets 1393
 - Example: ACLs and Bridged Packets 1394
 - Example: ACLs and Routed Packets 1394
 - Example: ACLs and Multicast Packets 1395

CHAPTER 71

Configuring DHCP 1397

- Restrictions for DHCP 1397
- Information About DHCP 1397

DHCP Server	1397
DHCP Relay Agent	1397
DHCP Snooping	1398
Option-82 Data Insertion	1399
Cisco IOS DHCP Server Database	1402
DHCP Snooping Binding Database	1402
DHCP Snooping and Switch Stacks	1403
How to Configure DHCP Features	1404
Default DHCP Snooping Configuration	1404
DHCP Snooping Configuration Guidelines	1405
Configuring the DHCP Server	1405
DHCP Server and Switch Stacks	1405
Configuring the DHCP Relay Agent	1405
Specifying the Packet Forwarding Address	1406
Prerequisites for Configuring DHCP Snooping and Option 82	1408
Enabling DHCP Snooping and Option 82	1409
Enabling the Cisco IOS DHCP Server Database	1412
Monitoring DHCP Snooping Information	1413
Configuring DHCP Server Port-Based Address Allocation	1413
Information About Configuring DHCP Server Port-Based Address Allocation	1413
Default Port-Based Address Allocation Configuration	1414
Port-Based Address Allocation Configuration Guidelines	1414
Enabling the DHCP Snooping Binding Database Agent	1414
Enabling DHCP Server Port-Based Address Allocation	1416
Monitoring DHCP Server Port-Based Address Allocation	1418
Additional References	1418
Feature Information for DHCP Snooping and Option 82	1418

CHAPTER 72

Configuring IP Source Guard	1421
Information About IP Source Guard	1421
IP Source Guard	1421
IP Source Guard for Static Hosts	1421
IP Source Guard Configuration Guidelines	1422
How to Configure IP Source Guard	1423

Enabling IP Source Guard	1423
Configuring IP Source Guard for Static Hosts on a Layer 2 Access Port	1424
Monitoring IP Source Guard	1426
Additional References	1427

CHAPTER 73

Configuring Dynamic ARP Inspection	1429
Restrictions for Dynamic ARP Inspection	1429
Understanding Dynamic ARP Inspection	1430
Interface Trust States and Network Security	1432
Rate Limiting of ARP Packets	1433
Relative Priority of ARP ACLs and DHCP Snooping Entries	1433
Logging of Dropped Packets	1433
Default Dynamic ARP Inspection Configuration	1434
Relative Priority of ARP ACLs and DHCP Snooping Entries	1434
Configuring ARP ACLs for Non-DHCP Environments	1434
Configuring Dynamic ARP Inspection in DHCP Environments	1437
Limiting the Rate of Incoming ARP Packets	1439
Performing Dynamic ARP Inspection Validation Checks	1442
Monitoring DAI	1443
Verifying the DAI Configuration	1444
Additional References	1444

CHAPTER 74

Configuring IEEE 802.1x Port-Based Authentication	1447
Information About 802.1x Port-Based Authentication	1447
Port-Based Authentication Process	1448
Port-Based Authentication Initiation and Message Exchange	1450
Authentication Manager for Port-Based Authentication	1451
Port-Based Authentication Methods	1451
Per-User ACLs and Filter-Ids	1452
Port-Based Authentication Manager CLI Commands	1453
Ports in Authorized and Unauthorized States	1454
Port-Based Authentication and Switch Stacks	1455
802.1x Host Mode	1456
802.1x Multiple Authentication Mode	1456

Multi-auth Per User VLAN assignment	1457
MAC Move	1458
MAC Replace	1459
802.1x Accounting	1459
802.1x Accounting Attribute-Value Pairs	1460
802.1x Readiness Check	1461
Switch-to-RADIUS-Server Communication	1461
802.1x Authentication with VLAN Assignment	1461
802.1x Authentication with Per-User ACLs	1463
802.1x Authentication with Downloadable ACLs and Redirect URLs	1464
Cisco Secure ACS and Attribute-Value Pairs for the Redirect URL	1465
Cisco Secure ACS and Attribute-Value Pairs for Downloadable ACLs	1466
VLAN ID-Based MAC Authentication	1466
802.1x Authentication with Guest VLAN	1466
802.1x Authentication with Restricted VLAN	1467
802.1x Authentication with Inaccessible Authentication Bypass	1468
Inaccessible Authentication Bypass Support on Multiple-Authentication Ports	1469
Inaccessible Authentication Bypass Authentication Results	1469
Inaccessible Authentication Bypass Feature Interactions	1469
802.1x Critical Voice VLAN	1470
802.1x User Distribution	1471
802.1x User Distribution Configuration Guidelines	1471
IEEE 802.1x Authentication with Voice VLAN Ports	1471
IEEE 802.1x Authentication with Port Security	1472
IEEE 802.1x Authentication with Wake-on-LAN	1472
IEEE 802.1x Authentication with MAC Authentication Bypass	1473
Network Admission Control Layer 2 IEEE 802.1x Validation	1474
Flexible Authentication Ordering	1474
Open1x Authentication	1475
Multidomain Authentication	1475
Limiting Login for Users	1477
802.1x Supplicant and Authenticator Switches with Network Edge Access Topology (NEAT)	1477
Voice Aware 802.1x Security	1478
Common Session ID	1479

How to Configure 802.1x Port-Based Authentication	1480
Default 802.1x Authentication Configuration	1480
802.1x Authentication Configuration Guidelines	1481
802.1x Authentication	1481
VLAN Assignment, Guest VLAN, Restricted VLAN, and Inaccessible Authentication Bypass	1482
MAC Authentication Bypass	1483
Maximum Number of Allowed Devices Per Port	1483
Configuring 802.1x Readiness Check	1483
Configuring Voice Aware 802.1x Security	1485
Configuring 802.1x Violation Modes	1487
Configuring 802.1x Authentication	1488
Configuring 802.1x Port-Based Authentication	1489
Configuring the Switch-to-RADIUS-Server Communication	1491
Configuring the Host Mode	1493
Configuring Periodic Re-Authentication	1494
Changing the Quiet Period	1496
Changing the Switch-to-Client Retransmission Time	1497
Setting the Switch-to-Client Frame-Retransmission Number	1498
Setting the Re-Authentication Number	1499
Enabling MAC Move	1500
Disabling MAC Move	1501
Enabling MAC Replace	1502
Configuring 802.1x Accounting	1504
Configuring a Guest VLAN	1505
Configuring a Restricted VLAN	1506
Configuring Number of Authentication Attempts on a Restricted VLAN	1508
Configuring 802.1x Inaccessible Authentication Bypass with Critical Voice VLAN	1509
Example of Configuring Inaccessible Authentication Bypass	1513
Configuring 802.1x Authentication with WoL	1513
Configuring MAC Authentication Bypass	1514
Formatting a MAC Authentication Bypass Username and Password	1515
Configuring 802.1x User Distribution	1516
Example of Configuring VLAN Groups	1517
Configuring NAC Layer 2 802.1x Validation	1518

Configuring Limiting Login for Users	1520
Configuring an Authenticator Switch with NEAT	1521
Configuring a Supplicant Switch with NEAT	1523
Configuring 802.1x Authentication with Downloadable ACLs and Redirect URLs	1526
Configuring Downloadable ACLs	1526
Configuring a Downloadable Policy	1527
Configuring VLAN ID-based MAC Authentication	1530
Configuring Flexible Authentication Ordering	1531
Configuring Open1x	1532
Disabling 802.1x Authentication on the Port	1534
Resetting the 802.1x Authentication Configuration to the Default Values	1535
Monitoring 802.1x Statistics and Status	1536
Additional References for IEEE 802.1x Port-Based Authentication	1536
Feature Information for 802.1x Port-Based Authentication	1537

CHAPTER 75
Configuring Web-Based Authentication 1539

Information About Web-Based Authentication	1539
Web-Based Authentication Overview	1539
Device Roles	1540
Host Detection	1541
Session Creation	1541
Authentication Process	1542
Using Authentication Proxy	1542
When to Use the Authentication Proxy	1543
Applying Authentication Proxy	1543
Local Web Authentication Banner	1544
Web Authentication Customizable Web Pages	1547
Web Authentication Redirection to Original URL Overview	1549
Web-based Authentication Interactions with Other Features	1551
Default Web-Based Authentication Configuration	1553
Web-Based Authentication Configuration Guidelines and Restrictions	1553
How to Configure Web-Based Authentication	1554
Configuring the Authentication Rule and Interfaces	1554
Configuring AAA Authentication	1556

Configuring Switch-to-RADIUS-Server Communication	1557
Configuring the HTTP Server	1559
Customizing the Authentication Proxy Web Pages	1560
Specifying a Redirection URL for Successful Login	1561
Configuring the Web-Based Authentication Parameters	1562
Configuring a Web Authentication Local Banner	1563
Configuring Web-Based Authentication without SVI	1564
Configuring Web-Based Authentication with VRF Aware	1565
Removing Web-Based Authentication Cache Entries	1567
Monitoring Web-Based Authentication Status	1567
Displaying Web-Based Authentication Status	1568
Monitoring HTTP Authentication Proxy	1568
Verifying HTTPS Authentication Proxy	1569
Configuration Examples for Web-Based Authentication	1570
Example: Configuring the Authentication Rule and Interfaces	1570
Example: AAA Configuration	1570
Example: HTTP Server Configuration	1570
Example: Customizing the Authentication Proxy Web Pages	1571
Example: Specifying a Redirection URL for Successful Login	1571
Additional References for Web-Based Authentication	1572
Feature Information for Web-Based Authentication	1572

CHAPTER 76**Auto Identity 1573**

Auto Identity	1573
Information About Auto Identity	1573
Auto Identity Overview	1573
Auto Identity Global Template	1574
Auto Identity Interface Templates	1574
Auto Identity Built-in Policies	1575
Auto Identity Class Maps Templates	1575
Auto Identity Parameter Maps	1576
Auto Identity Service Templates	1576
How to Configure Auto Identity	1576
Configuring Auto Identity Globally	1576

Configuring Auto Identity at an Interface Level	1578
Configuration Examples for Auto Identity	1579
Example: Configuring Auto Identity Globally	1579
Example: Configuring Auto Identity at an Interface Level	1579
Verifying Auto Identity	1579
Feature Information for Auto Identity	1583

CHAPTER 77

Configuring Port-Based Traffic Control	1585
Overview of Port-Based Traffic Control	1586
Finding Feature Information	1586
Information About Storm Control	1586
Storm Control	1586
How Traffic Activity is Measured	1586
Traffic Patterns	1587
How to Configure Storm Control	1588
Configuring Storm Control and Threshold Levels	1588
Configuring Small-Frame Arrival Rate	1590
Finding Feature Information	1592
Information About Protected Ports	1593
Protected Ports	1593
Default Protected Port Configuration	1593
Protected Ports Guidelines	1593
How to Configure Protected Ports	1593
Configuring a Protected Port	1593
Monitoring Protected Ports	1595
Where to Go Next	1595
Additional References	1595
Feature Information	1596
Finding Feature Information	1596
Information About Port Blocking	1596
Port Blocking	1596
How to Configure Port Blocking	1597
Blocking Flooded Traffic on an Interface	1597
Monitoring Port Blocking	1598

Where to Go Next	1599
Additional References	1599
Feature Information	1600
Prerequisites for Port Security	1600
Restrictions for Port Security	1600
Information About Port Security	1600
Port Security	1600
Types of Secure MAC Addresses	1600
Sticky Secure MAC Addresses	1601
Security Violations	1601
Port Security Aging	1602
Port Security and Switch Stacks	1602
Default Port Security Configuration	1603
Port Security Configuration Guidelines	1603
Overview of Port-Based Traffic Control	1604
How to Configure Port Security	1605
Enabling and Configuring Port Security	1605
Enabling and Configuring Port Security Aging	1610
Configuration Examples for Port Security	1612
Additional References	1612
Finding Feature Information	1613
Information About Protocol Storm Protection	1613
Protocol Storm Protection	1613
Default Protocol Storm Protection Configuration	1614
How to Configure Protocol Storm Protection	1614
Enabling Protocol Storm Protection	1614
Monitoring Protocol Storm Protection	1615
Additional References	1615
<hr/>	
CHAPTER 78	Configuring Cisco TrustSec 1617
	Information about Cisco TrustSec 1617
	Cisco TrustSec Features 1618
	Feature Information for Cisco TrustSec 1618

CHAPTER 79	Configuring FIPS	1619
	Information About FIPS and Common Criteria	1619

CHAPTER 80	Configuring Control Plane Policing	1621
	Restrictions for Control Plane Policing	1621
	Control Plane Policing	1621
	Configuring Control Plane Policing	1622
	Examples: Configuring CoPP	1623

PART XII	Configuring Cisco IOS IP SLAs	1625
-----------------	--------------------------------------	-------------

CHAPTER 81	Configuring Cisco IP SLAs	1627
	Restrictions on SLAs	1627
	Information About SLAs	1627
	Cisco IOS IP Service Level Agreements (SLAs)	1627
	Network Performance Measurement with Cisco IOS IP SLAs	1628
	IP SLA Responder and IP SLA Control Protocol	1629
	Response Time Computation for IP SLAs	1630
	How to Configure IP SLAs Operations	1631
	Default Configuration	1631
	Configuration Guidelines	1631
	Configuring the IP SLA Responder	1631
	Monitoring IP SLA Operations	1632
	Additional References	1633
	Feature History and Information for Service Level Agreements	1634

PART XIII	Stacking	1635
------------------	-----------------	-------------

CHAPTER 82	Managing Switch Stacks	1637
	Prerequisites for Switch Stacks	1637
	Restrictions for Switch Stacks	1637
	Information About Switch Stacks	1638
	Switch Stack Overview	1638

Supported Features in a Switch Stack	1638
Switch Stack Membership	1639
Changes to Switch Stack Membership	1639
Stack Member Numbers	1641
Stack Member Priority Values	1642
Switch Stack Bridge ID and MAC Address	1642
Persistent MAC Address on the Switch Stack	1642
Active and Standby Switch Election and Reelection	1643
Switch Stack Configuration Files	1643
Offline Configuration to Provision a Stack Member	1644
Effects of Adding a Provisioned Switch to a Switch Stack	1645
Effects of Replacing a Provisioned Switch in a Switch Stack	1646
Effects of Removing a Provisioned Switch from a Switch Stack	1646
Stack Protocol Version	1646
Major Stack Protocol Version Number Incompatibility Among Stack-Capable Switches	1646
Minor Stack Protocol Version Number Incompatibility Among Stack-Capable Switches	1647
Auto-Upgrade	1647
Auto-Advise	1648
SDM Template Mismatch in Switch Stacks	1650
Switch Stack Management Connectivity	1650
Connectivity to Specific Stack Members	1650
Connectivity to the Switch Stack Through an IP Address	1650
Connectivity to the Switch Stack Through Console Ports or Ethernet Management Ports	1651
How to Configure a Switch Stack	1651
Enabling the Persistent MAC Address Feature	1651
Assigning a Stack Member Number	1653
Setting the Stack Member Priority Value	1654
Provisioning a New Member for a Switch Stack	1655
Removing Provisioned Switch Information	1656
Troubleshooting the Switch Stack	1657
Accessing the CLI of a Specific Member	1657
Temporarily Disabling a Stack Port	1657
Reenabling a Stack Port While Another Member Starts	1658
Monitoring the Device Stack	1659

Configuration Examples for Switch Stacks	1660
Switch Stack Configuration Scenarios	1660
Enabling the Persistent MAC Address Feature: Example	1661
Provisioning a New Member for a Switch Stack: Example	1662
Additional References for Switch Stacks	1662

CHAPTER 83**FlexStack-Extended 1665**

Restrictions for FlexStack-Extended	1665
Information About FlexStack-Extended	1665
FlexStack-Extended	1665
FlexStack-Extended on Catalyst 2960-X and 2960-XR Switches	1666
Default Port Configurations	1667
FlexStack-Extended LED	1668
How to Configure FlexStack-Extended	1668
Configuring a Stack Port as a Network Port	1668
Configuring a Network Port as a Stack Port	1669
Configuring the Stack Speed	1671
Configuration Examples for FlexStack-Extended	1671
Examples: Configuring FlexStack-Extended	1671
Feature Information for FlexStack-Extended	1672

PART XIV**VLAN 1673**

CHAPTER 84**Configuring VTP 1675**

Prerequisites for VTP	1675
Information About VTP	1676
VTP	1676
VTP Domain	1676
VTP Modes	1677
VTP Advertisements	1678
VTP Version 2	1679
VTP Version 3	1679
VTP Pruning	1680
VTP and Switch Stacks	1680

VTP Configuration Guidelines	1681
Configuration Requirements	1681
VTP Settings	1681
Domain Names for Configuring VTP	1682
Passwords for the VTP Domain	1682
VTP Version	1682
Default VTP Configuration	1683
How to Configure VTP	1684
Configuring VTP Mode	1684
Configuring a VTP Version 3 Password	1686
Configuring a VTP Version 3 Primary Server	1688
Enabling the VTP Version	1688
Enabling VTP Pruning	1690
Configuring VTP on a Per-Port Basis	1691
Adding a VTP Client to a VTP Domain	1692
Monitoring VTP	1694
Configuration Examples for VTP	1695
Example: Configuring the Switch as a VTP Server	1695
Example: Configuring a Hidden Password	1695
Example: Configuring a VTP Version 3 Primary Server	1696
Example: Configuring VTP on a Per-Port Basis	1696
Where to Go Next	1696
Additional References	1697
Feature History and Information for VTP	1697

CHAPTER 85**VLANs 1699**

Prerequisites for VLANs	1699
Restrictions for VLANs	1699
Information About VLANs	1699
Logical Networks	1699
Supported VLANs	1700
VLAN Port Membership Modes	1700
VLAN Configuration Files	1701
Normal-Range VLAN Overview	1702

Token Ring VLANs	1702
Normal-Range VLANs Configuration Process	1702
VLAN Configuration Saving Process	1703
Normal-Range VLAN Configuration Guidelines	1703
Extended-Range VLAN Configuration Guidelines	1704
Default Ethernet VLAN Configuration	1705
Default VLAN Configuration	1705
How to Configure VLANs	1706
How to Configure Normal-Range VLANs	1706
Creating or Modifying an Ethernet VLAN	1706
Deleting a VLAN	1708
Assigning Static-Access Ports to a VLAN	1709
How to Configure Extended-Range VLANs	1711
Creating an Extended-Range VLAN	1711
Creating an Extended-Range VLAN with an Internal VLAN ID	1713
Monitoring VLANs	1715
Configuration Examples	1716
Example: Creating a VLAN Name	1716
Example: Configuring a Port as Access Port	1716
Example: Creating an Extended-Range VLAN	1716
Where to Go Next	1716
Additional References	1717
Feature History and Information for VLAN	1717

CHAPTER 86**Configuring VLAN Trunks 1719**

Prerequisites for VLAN Trunks	1719
Restrictions for VLAN Trunks	1719
Information About VLAN Trunks	1720
Trunking Overview	1720
Trunking Modes	1721
Layer 2 Interface Modes	1721
Allowed VLANs on a Trunk	1722
Load Sharing on Trunk Ports	1722
Network Load Sharing Using STP Priorities	1722

Network Load Sharing Using STP Path Cost	1722
Feature Interactions	1723
Default Layer 2 Ethernet Interface VLAN Configuration	1723
How to Configure VLAN Trunks	1724
Configuring an Ethernet Interface as a Trunk Port	1724
Configuring a Trunk Port	1724
Defining the Allowed VLANs on a Trunk	1726
Changing the Pruning-Eligible List	1727
Configuring the Native VLAN for Untagged Traffic	1729
Configuring Trunk Ports for Load Sharing	1730
Configuring Load Sharing Using STP Port Priorities	1730
Configuring Load Sharing Using STP Path Cost	1734
Configuration Examples for VLAN Trunking	1737
Example: Configuring an IEEE 802.1Q Trunk	1737
Example: Removing a VLAN	1737
Where to Go Next	1737
Additional References	1738
Feature History and Information for VLAN Trunks	1738
<hr/>	
CHAPTER 87	Configuring Private VLANs 1739
Prerequisites for Private VLANs	1739
Secondary and Primary VLAN Configuration	1739
Private VLAN Port Configuration	1741
Restrictions for Private VLANs	1742
Limitations with Other Features	1742
Information About Private VLANs	1743
Private VLAN Domains	1743
Secondary VLANs	1744
Private VLANs Ports	1744
Private VLANs in Networks	1745
IP Addressing Scheme with Private VLANs	1745
Private VLANs Across Multiple Devices	1746
Private VLAN Interaction with Other Features	1746
Private VLANs and Unicast, Broadcast, and Multicast Traffic	1746

Private VLANs and SVIs	1747
Private VLANs and Device Stacks	1747
Private VLAN Configuration Tasks	1747
Default Private VLAN Configuration	1748
How to Configure Private VLANs	1748
Configuring and Associating VLANs in a Private VLAN	1748
Configuring a Layer 2 Interface as a Private VLAN Host Port	1752
Configuring a Layer 2 Interface as a Private VLAN Promiscuous Port	1753
Mapping Secondary VLANs to a Primary VLAN Layer 3 VLAN Interface	1755
Monitoring Private VLANs	1757
Configuration Examples for Private VLANs	1757
Example: Configuring a Primary VLAN, Isolated VLAN, and a Community of VLANs	1757
Example: Configuring an Interface as a Host Port	1758
Example: Configuring an Interface as a Private VLAN Promiscuous Port	1758
Example: Mapping Secondary VLANs to a Primary VLAN Interface	1759
Example: Monitoring Private VLANs	1759
Where to Go Next	1759
Additional References	1760
Feature History and Information for Private VLANs	1760

CHAPTER 88

Configuring VMPS	1761
Prerequisites for VMPS	1761
Restrictions for VMPS	1761
Information About VMPS	1762
Dynamic VLAN Assignments	1762
Dynamic-Access Port VLAN Membership	1763
Default VMPS Client Configuration	1763
How to Configure VMPS	1764
Entering the IP Address of the VMPS	1764
Configuring Dynamic-Access Ports on VMPS Clients	1765
Reconfirming VLAN Memberships	1767
Changing the Reconfirmation Interval	1768
Changing the Retry Count	1769
Troubleshooting Dynamic-Access Port VLAN Membership	1770

Monitoring the VMPS	1770
Configuration Example for VMPS	1771
Example: VMPS Configuration	1771
Where to Go Next	1772
Additional References	1773
Feature History and Information for VMPS	1774

CHAPTER 89

Configuring IEEE 802.1Q and Layer 2 Protocol Tunneling	1775
Prerequisites for Configuring Tunneling	1775
IEEE 802.1Q Tunneling	1775
Layer 2 Protocol Tunneling	1776
Layer 2 Tunneling for EtherChannels	1777
Information about Tunneling	1777
IEEE 802.1Q and Layer 2 Protocol Overview	1777
IEEE 802.1Q Tunneling	1777
IEEE 802.1Q Tunneling Configuration Guidelines	1779
Native VLANs	1780
System MTU	1781
Default IEEE 802.1Q Tunneling Configuration	1781
Layer 2 Protocol Tunneling Overview	1781
Layer 2 Protocol Tunneling on Ports	1783
Default Layer 2 Protocol Tunneling Configuration	1784
How to Configure Tunneling	1785
Configuring an IEEE 802.1Q Tunneling Port	1785
Configuring Layer 2 Protocol Tunneling	1787
Configuring the SP Edge Switch	1790
Configuring the Customer Device	1793
Configuration Examples for IEEE 802.1Q and Layer 2 Protocol Tunneling	1796
Example: Configuring an IEEE 802.1Q Tunneling Port	1796
Example: Configuring Layer 2 Protocol Tunneling	1796
Examples: Configuring the SP Edge and Customer Switches	1797
Monitoring Tunneling Status	1798
Where to Go Next	1799
Additional References	1799

Feature History and Information for Tunneling 1800

CHAPTER 90

Configuring Voice VLANs 1801

Prerequisites for Voice VLANs 1801

Restrictions for Voice VLANs 1801

Information About Voice VLAN 1802

Voice VLANs 1802

Cisco IP Phone Voice Traffic 1802

Cisco IP Phone Data Traffic 1802

Voice VLAN Configuration Guidelines 1803

Default Voice VLAN Configuration 1804

How to Configure Voice VLAN 1804

Configuring Cisco IP Phone Voice Traffic 1804

Configuring the Priority of Incoming Data Frames 1806

Monitoring Voice VLAN 1808

Configuration Examples for Voice VLANs 1808

Example: Configuring Cisco IP Phone Voice Traffic 1808

Example: Configuring a Port Connected to an IP Phone Not to Change Frame Priority 1809

Where to Go Next 1809

Additional References 1809

Feature History and Information for Voice VLAN 1810

PART XV

System Management 1811

CHAPTER 91

Administering the System 1813

Information About Administering the Device 1813

System Time and Date Management 1813

System Clock 1813

Real Time Clock 1814

Network Time Protocol 1814

NTP Stratum 1815

NTP Associations 1816

NTP Security 1816

NTP Implementation 1816

NTP Version 4	1817
System Name and Prompt	1817
Stack System Name and Prompt	1817
Default System Name and Prompt Configuration	1817
DNS	1818
Default DNS Settings	1818
Login Banners	1818
Default Banner Configuration	1818
MAC Address Table	1818
MAC Address Table Creation	1819
MAC Addresses and VLANs	1819
MAC Addresses and Device Stacks	1819
Default MAC Address Table Settings	1820
ARP Table Management	1820
How to Administer the Device	1820
Configuring the Time and Date Manually	1820
Setting the System Clock	1820
Configuring the Time Zone	1821
Configuring Summer Time (Daylight Saving Time)	1822
Configuring a System Name	1826
Setting Up DNS	1827
Configuring a Message-of-the-Day Login Banner	1828
Configuring a Login Banner	1830
Managing the MAC Address Table	1831
Changing the Address Aging Time	1831
Configuring MAC Address Change Notification Traps	1832
Configuring MAC Address Move Notification Traps	1834
Configuring MAC Threshold Notification Traps	1836
Adding and Removing Static Address Entries	1838
Configuring Unicast MAC Address Filtering	1839
Monitoring and Maintaining Administration of the Device	1841
Configuration Examples for Device Administration	1842
Example: Setting the System Clock	1842
Examples: Configuring Summer Time	1842

Example: Configuring a MOTD Banner	1842
Example: Configuring a Login Banner	1843
Example: Configuring MAC Address Change Notification Traps	1843
Example: Configuring MAC Threshold Notification Traps	1843
Example: Adding the Static Address to the MAC Address Table	1843
Example: Configuring Unicast MAC Address Filtering	1844
Additional References for Switch Administration	1844
Feature History and Information for Device Administration	1845

CHAPTER 92**Performing Device Setup Configuration 1847**

Information About Performing Device Setup Configuration	1847
Boot Process	1847
Devices Information Assignment	1848
Default Switch Information	1848
DHCP-Based Autoconfiguration Overview	1849
DHCP Client Request Process	1849
DHCP-based Autoconfiguration and Image Update	1850
Restrictions for DHCP-based Autoconfiguration	1851
DHCP Autoconfiguration	1851
DHCP Auto-Image Update	1851
DHCP Server Configuration Guidelines	1851
Purpose of the TFTP Server	1852
Purpose of the DNS Server	1853
How to Obtain Configuration Files	1853
How to Control Environment Variables	1854
Common Environment Variables	1855
Environment Variables for TFTP	1857
Scheduled Reload of the Software Image	1857
How to Perform Device Setup Configuration	1858
Configuring DHCP Autoconfiguration (Only Configuration File)	1858
Configuring DHCP Auto-Image Update (Configuration File and Image)	1860
Configuring the Client to Download Files from DHCP Server	1863
Manually Assigning IP Information to Multiple SVIs	1864
Configuring the NVRAM Buffer Size	1866

Modifying the Device Startup Configuration	1867
Specifying the Filename to Read and Write the System Configuration	1867
Manually Booting the Switch	1868
Configuring a Scheduled Software Image Reload	1869
Monitoring Device Setup Configuration	1870
Example: Verifying the Device Running Configuration	1870
Examples: Displaying Software Install	1871
Configuration Examples for Performing Device Setup	1871
Example: Configuring a Device as a DHCP Server	1871
Example: Configuring DHCP Auto-Image Update	1872
Example: Configuring a Device to Download Configurations from a DHCP Server	1872
Example: Configuring NVRAM Buffer Size	1873
Additional References for Performing Switch Setup	1873
Feature History and Information For Performing Device Setup Configuration	1874

CHAPTER 93

Configuring AVC with DNS-AS	1875
Prerequisites for AVC with DNS-AS	1875
Restrictions and Guidelines for AVC with DNS-AS	1875
Information About AVC with DNS-AS	1876
Overview of AVC with DNS-AS	1876
Key Concepts for AVC with DNS-AS	1877
AVC with DNS-AS Process Flow	1878
DNS Snooping Process	1878
DNS-AS Client Process	1878
Figure: AVC with DNS-AS Process Flow	1879
Stacking and AVC with DNS-AS	1880
Default Configuration for AVC with DNS-AS	1880
How to Configure AVC with DNS-AS	1880
Generating Metadata Streams	1880
Configuring a DNS Server as the Authoritative Server	1882
Enabling AVC with DNS-AS	1883
Maintaining the List of Trusted Domains	1883
Configuring QoS for AVC with DNS-AS	1884
Configuring FNF for AVC with DNS-AS	1887

Option Templates	1888
Sample FNF Configuration for AVC with DNS-AS	1890
Monitoring AVC with DNS-AS	1893
Troubleshooting AVC with DNS-AS	1896
Feature History and Information for AVC with DNS-AS	1897

CHAPTER 94**Configuring SDM Templates 1899**

Information About Configuring SDM Templates	1899
Restrictions for SDM Templates	1899
SDM Templates	1899
SDM Template Resources	1900
SDM Templates and Switch Stacks	1901
How to Configure SDM Templates	1901
Setting the SDM Template	1901
Configuration Examples for SDM Templates	1903
Examples: Displaying SDM Templates	1903
Examples: Configuring SDM Templates	1904
Additional References for SDM Templates	1904
Feature History and Information for Configuring SDM Templates	1905

CHAPTER 95**Configuring System Message Logging and Smart Logging 1907**

Information About Configuring System Message Logs and Smart Logs	1907
System Message Logging	1907
System Log Message Format	1908
Default System Message Logging Settings	1909
Syslog Message Limits	1909
Smart Logging	1909
Smart Logging for Port ACL Deny or Permit Actions	1910
How to Configure System Message Logs and Smart Logs	1910
Setting the Message Display Destination Device	1910
Synchronizing Log Messages	1912
Disabling Message Logging	1913
Enabling and Disabling Time Stamps on Log Messages	1914
Enabling and Disabling Sequence Numbers in Log Messages	1915

Defining the Message Severity Level	1916
Limiting Syslog Messages Sent to the History Table and to SNMP	1917
Logging Messages to a UNIX Syslog Daemon	1918
Enabling Smart Logging	1919
Enabling Smart Logging for DHCP Snooping Violations	1920
Enabling Smart Logging for Dynamic ARP Inspection Violations	1920
Enabling Smart Logging for IP Source Guard Violations	1921
Monitoring and Maintaining System Message Logs and Smart Logs	1922
Monitoring Configuration Archive Logs	1922
Monitoring Smart Logging	1922
Configuration Examples for System Message Logs and Smart Logs	1923
Example: Stacking System Message	1923
Example: Switch System Message	1923
Example: Enabling Smart Logging	1923
Examples: Displaying Service Timestamps Log	1924
Additional References for System Message Logs and Smart Logs	1924
Feature History and Information For System Message Logs	1925

CHAPTER 96
Configuring Online Diagnostics 1927

Information About Configuring Online Diagnostics	1927
Online Diagnostics	1927
How to Configure Online Diagnostics	1928
Starting Online Diagnostic Tests	1928
Configuring Online Diagnostics	1928
Scheduling Online Diagnostics	1928
Configuring Health-Monitoring Diagnostics	1929
Monitoring and Maintaining Online Diagnostics	1932
Displaying Online Diagnostic Tests and Test Results	1932
Configuration Examples for Online Diagnostic Tests	1933
Starting Online Diagnostic Tests	1933
Example: Configure a Health Monitoring Test	1933
Examples: Schedule Diagnostic Test	1934
Displaying Online Diagnostics: Examples	1934
Additional References for Online Diagnostics	1936

Feature History and Information for Configuring Online Diagnostics 1937

CHAPTER 97

Troubleshooting the Software Configuration 1939

Information About Troubleshooting the Software Configuration 1939

Software Failure on a Switch 1939

Lost or Forgotten Password on a Device 1939

Power over Ethernet Ports 1940

Disabled Port Caused by Power Loss 1940

Disabled Port Caused by False Link-Up 1940

Ping 1941

Layer 2 Traceroute 1941

Layer 2 Traceroute Guidelines 1941

IP Traceroute 1942

Time Domain Reflector Guidelines 1943

Debug Commands 1944

Onboard Failure Logging on the Switch 1944

Possible Symptoms of High CPU Utilization 1944

How to Troubleshoot the Software Configuration 1945

Recovering from a Software Failure 1945

Recovering from a Lost or Forgotten Password 1947

Procedure with Password Recovery Enabled 1948

Procedure with Password Recovery Disabled 1950

Recovering from a Command Switch Failure 1951

Replacing a Failed Command Switch with a Cluster Member 1952

Replacing a Failed Command Switch with Another Switch 1954

Preventing Switch Stack Problems 1955

Preventing Autonegotiation Mismatches 1956

Troubleshooting SFP Module Security and Identification 1956

Monitoring SFP Module Status 1957

Executing Ping 1957

Monitoring Temperature 1957

Monitoring the Physical Path 1957

Executing IP Traceroute 1958

Running TDR and Displaying the Results 1958

Redirecting Debug and Error Message Output	1958
Using the show platform forward Command	1959
Configuring OBFL	1959
Verifying Troubleshooting of the Software Configuration	1959
Displaying OBFL Information	1959
Example: Verifying the Problem and Cause for High CPU Utilization	1961
Scenarios for Troubleshooting the Software Configuration	1963
Scenarios to Troubleshoot Power over Ethernet (PoE)	1963
Configuration Examples for Troubleshooting Software	1965
Example: Pinging an IP Host	1965
Example: Performing a Traceroute to an IP Host	1966
Example: Enabling All System Diagnostics	1967
Additional References for Troubleshooting Software Configuration	1967
Feature History and Information for Troubleshooting Software Configuration	1968

CHAPTER 98**Information About Licensing 1969**

Restrictions for Configuring Licenses	1969
Information About Licensing	1969
Overview of License Levels	1969
Base Licenses	1970
Add-On Licenses	1970
License States	1970
Guidelines for License Types	1971
Ordering with Smart Accounts	1971
License Activation for Switch Stacks	1971
How to Configure Add-On License Levels	1972
Activating an Image Based Add-on License	1972
Rehosting a License	1973
Monitoring Licenses	1974
Configuration Examples for License Levels	1974
Reference	1974
Example: Displaying the detailed license information	1974
Example: Displaying a summary of the license information	1975
Example: Displaying the end user license agreement	1975

Feature History for Information About Licensing 1976

PART XVI

Embedded Event Manager 1977

CHAPTER 99

Embedded Event Manager Overview 1979

Information About Embedded Event Manager 1979

Embedded Event Manager 1979

Embedded Event Manager 1.0 1980

Embedded Event Manager 2.0 1981

Embedded Event Manager 2.1 1981

Embedded Event Manager 2.1 (Software Modularity) 1982

Embedded Event Manager 2.2 1982

Embedded Event Manager 2.3 1983

Embedded Event Manager 2.4 1983

Embedded Event Manager 3.0 1984

Embedded Event Manager 3.1 1985

Embedded Event Manager 3.2 1985

Embedded Event Manager 4.0 1986

EEM Event Detectors Available by Cisco IOS Release 1987

Event Detectors 1989

EEM Actions Available by Cisco IOS Release 1993

Embedded Event Manager Actions 1994

Embedded Event Manager Environment Variables 1994

Embedded Event Manager Policy Creation 1996

Where to Go Next 1997

Feature Information for Embedded Event Manager 4.0 Overview 1997

Additional References 1997

CHAPTER 100

Information About Writing EEM Policies Using the Cisco IOS CLI 2001

Prerequisites for Writing EEM Policies Using the Cisco IOS CLI 2001

Information About Writing EEM Policies Using the Cisco IOS CLI 2001

Embedded Event Manager Policies 2001

EEM Applet 2002

EEM Script 2002

Embedded Event Manager Built-In Environment Variables Used in EEM Applets	2002
How to Write EEM Policies Using the Cisco IOS CLI	2013
Registering and Defining an Embedded Event Manager Applet	2013
EEM Environment Variables	2013
Alphabetical Order of EEM Action Labels	2014
Troubleshooting Tips	2017
Registering and Defining an EEM Tcl Script	2017
Unregistering Embedded Event Manager Policies	2019
Suspending All Embedded Event Manager Policy Execution	2020
Displaying Embedded Event Manager History Data	2021
Displaying Embedded Event Manager Registered Policies	2023
Configuring Event SNMP Notification	2024
Configuring Multiple Event Support	2025
Setting the Event Configuration Parameters	2025
Configuring EEM Class-Based Scheduling	2027
Holding a Scheduled EEM Policy Event or Event Queue	2028
Resuming Execution of EEM Policy Events or Event Queues	2029
Clearing Pending EEM Policy Events or Event Queues	2030
Modifying the Scheduling Parameters of EEM Policy Events or Event Queues	2032
Verifying Class-Based Active EEM Policies	2033
Verifying Class-Based Active EEM Policies	2034
Verifying Pending EEM Policies	2034
Configuring EEM Applet (Interactive CLI) Support	2035
Reading and Writing Input from the Active Console for Synchronous EEM Applets	2035
Configuring SNMP Library Extensions	2038
Prerequisites	2038
SNMP Get and Set Operations	2038
SNMP Traps and Inform Requests	2041
Configuring EEM Applet for SNMP Get and Set Operations	2041
Configuring EEM Applet for SNMP OID Notifications	2043
Configuring Variable Logic for EEM Applets	2046
Prerequisites	2046
Configuring Variable Logic for EEM Applets	2046
Specifying a Loop of Conditional Blocks	2047

Specifying if else Conditional Blocks	2048
Specifying foreach Iterating Statements	2049
Using Regular Expressions	2051
Incrementing the Values of Variables	2052
Configuring Event SNMP Object	2052
Disabling AAA Authorization	2054
Configuring Description of an Embedded Event Manager Applet	2055
Configuration Examples for Writing Embedded Event Manager Policies Using Tcl	2056
Embedded Event Manager Applet Configuration Examples	2056
Configuration Examples for Embedded Event Manager Applet	2061
Example Identity Event Detector	2061
Example MAT Event Detector	2061
Example Neighbor-Discovery Event Detector	2061
Embedded Event Manager Manual Policy Execution Examples	2062
Embedded Event Manager Watchdog System Monitor (Cisco IOS) Event Detector Configuration Example	2062
Configuration SNMP Library Extensions Examples	2063
SNMP Get Operations Examples	2063
SNMP GetID Operations Examples	2064
Set Operations Examples	2065
Generating SNMP Notifications Examples	2066
Configuring Variable Logic for EEM Applets Examples	2067
Configuring Event SNMP-Object Examples	2072
Configuring Description of an EEM Applet Examples	2072
Additional References	2072
Feature Information for Writing EEM 4.0 Policies Using the Cisco IOS CLI	2074

CHAPTER 101**Writing Embedded Event Manager Policies Using Tcl 2075**

Prerequisites for Writing Embedded Event Manager Policies Using Tcl	2075
Information About Writing Embedded Event Manager Policies Using Tcl	2075
EEM Policies	2075
EEM Policy Tcl Command Extension Categories	2077
General Flow of EEM Event Detection and Recovery	2077
Safe-Tcl	2078

Bytecode Support for EEM 2.4	2080
Registration Substitution	2080
Cisco File Naming Convention for EEM	2081
How to Write Embedded Event Manager Policies Using Tcl	2082
Registering and Defining an EEM Tcl Script	2082
Displaying EEM Registered Policies	2084
Unregistering EEM Policies	2085
Suspending EEM Policy Execution	2087
Managing EEM Policies	2088
Modifying History Table Size and Displaying EEM History Data	2090
Displaying Software Modularity Process Reliability Metrics Using EEM	2091
Troubleshooting Tips	2092
Modifying the Sample EEM Policies	2092
Sample EEM Policies	2093
Programming EEM Policies with Tcl	2095
Tcl Policy Structure and Requirements	2095
EEM Entry Status	2097
EEM Exit Status	2097
EEM Policies and Cisco Error Number	2098
Troubleshooting Tips	2104
Creating an EEM User Tcl Library Index	2105
Creating an EEM User Tcl Package Index	2108
Configuration Examples for Writing Embedded Event Manager Policies Using Tcl	2111
Assigning a Username for a Tcl Session Examples	2111
EEM Event Detector Demo Examples	2111
Programming Policies with Tcl Sample Scripts Example	2119
Debugging Embedded Event Manager Policies Examples	2128
Tracing Tcl set Command Operations Example	2130
RPC Event Detector Example	2130
Additional References	2132

CHAPTER 102**Signed Tcl Scripts 2135**

Prerequisites for Signed Tcl Scripts	2135
Restrictions for Signed Tcl Scripts	2135

Information About Signed Tcl Scripts	2136
Cisco PKI	2136
RSA Key Pair	2136
Certificate and Trustpoint	2137
How to Configure Signed Tcl Scripts	2137
Generating a Key Pair	2137
Generating a Certificate	2138
Signing the Tcl Scripts	2140
Verifying the Signature	2140
Converting the Signature into Nonbinary Data	2141
Configuring the Device with a Certificate	2144
Verifying the Trustpoint	2148
Verifying the Signed Tcl Script	2148
What to Do Next	2149
Configuration Examples for Signed Tcl Script	2150
Generating a Key Pair Example	2150
Generating a Certificate Example	2150
Signing the Tcl Scripts Example	2151
Verifying the Signature Example	2151
Converting the Signature with Nonbinary Data Example	2151
Configuring the Device with a Certificate Example	2153
Additional References	2154
Feature Information for Signed Tcl Scripts	2155
Glossary	2155
Notices	2156
OpenSSL Open SSL Project	2156
License Issues	2156

CHAPTER 103	EEM CLI Library Command Extensions	2159
	cli_close	2160
	cli_exec	2160
	cli_get_ttyname	2161
	cli_open	2161
	cli_read	2162

cli_read_drain	2162
cli_read_line	2163
cli_read_pattern	2163
cli_run	2164
cli_run_interactive	2165
cli_write	2166
EEM 4.0 CLI Library XML-PI Support	2169
EEM CLI Library XML-PI Support	2169

CHAPTER 104 **EEM Context Library Command Extensions** 2171

context_retrieve	2171
context_save	2174

CHAPTER 105 **EEM Event Registration Tel Command Extensions** 2179

event_register_appl	2180
event_register_cli	2182
event_register_counter	2185
event_register_gold	2187
event_register_identity	2193
event_register_interface	2195
event_register_ioswdsysmon	2200
event_register_ipsla	2203
event_register_mat	2206
event_register_neighbor_discovery	2208
event_register_nf	2211
event_register_none	2214
event_register_oir	2216
event_register_process	2218
event_register_resource	2220
event_register_rf	2222
event_register_routing	2225
event_register_rpc	2227
event_register_snmp	2229
event_register_snmp_notification	2233

event_register_snmp_object 2235
 event_register_syslog 2238
 event_register_timer 2240
 event_register_timer_subscriber 2244
 event_register_track 2246
 event_register_wdysysmon 2248

CHAPTER 106 EEM Event Tcl Command Extensions 2263

event_completion 2263
 event_completion_with_wait 2264
 event_publish 2265
 event_wait 2268

CHAPTER 107 EEM Library Debug Command Extensions 2271

cli_debug 2271
 smtp_debug 2271

CHAPTER 108 EEM Multiple Event Support Tcl Command Extensions 2273

attribute 2273
 correlate 2274
 trigger 2275

CHAPTER 109 EEM SMTP Library Command Extensions 2277

smtp_send_email 2278
 smtp_subst 2279

CHAPTER 110 EEM System Information Tcl Command Extensions 2281

sys_reqinfo_cli_freq 2282
 sys_reqinfo_cli_history 2283
 sys_reqinfo_cpu_all 2283
 sys_reqinfo_crash_history 2284
 sys_reqinfo_mem_all 2285
 sys_reqinfo_proc 2286

sys_reqinfo_proc_all 2288
 sys_reqinfo_routename 2288
 sys_reqinfo_snmp 2289
 sys_reqinfo_syslog_freq 2290
 sys_reqinfo_syslog_history 2291

CHAPTER 111 EEM Utility Tel Command Extensions 2293

appl_read 2294
 appl_reqinfo 2294
 appl_setinfo 2295
 counter_modify 2296
 description 2297
 fts_get_stamp 2298
 register_counter 2299
 register_timer 2300
 timer_arm 2302
 timer_cancel 2303
 unregister_counter 2304

PART XVII Working with the Cisco IOS File System, Configuration Files, and Software Images 2307

CHAPTER 112 Working with the Cisco IOS File System, Configuration Files, and Software Images 2309

Working with the Flash File System 2309
 Information About the Flash File System 2309
 Displaying Available File Systems 2309
 Setting the Default File System 2312
 Displaying Information About Files on a File System 2312
 Changing Directories and Displaying the Working Directory 2313
 Creating Directories 2314
 Removing Directories 2314
 Copying Files 2315
 Copying Files from One Device in a Stack to Another Device in the Same Stack 2315
 Deleting Files 2316
 Creating, Displaying and Extracting Files 2316

Working with Configuration Files	2318
Information on Configuration Files	2318
Guidelines for Creating and Using Configuration Files	2319
Configuration File Types and Location	2319
Creating a Configuration File By Using a Text Editor	2319
Copying Configuration Files By Using TFTP	2320
Preparing to Download or Upload a Configuration File By Using TFTP	2320
Downloading the Configuration File By Using TFTP	2321
Uploading the Configuration File By Using TFTP	2321
Copying a Configuration File from the Device to an FTP Server	2322
Understanding the FTP Username and Password	2322
Preparing to Download or Upload a Configuration File By Using FTP	2323
Downloading a Configuration File By Using FTP	2323
Uploading a Configuration File By Using FTP	2325
Copying Configuration Files By Using RCP	2326
Preparing to Download or Upload a Configuration File By Using RCP	2326
Downloading a Configuration File By Using RCP	2327
Uploading a Configuration File By Using RCP	2328
Clearing Configuration Information	2329
Clearing the Startup Configuration File	2329
Deleting a Stored Configuration File	2329
Replacing and Rolling Back Configurations	2330
Information on Configuration Replacement and Rollback	2330
Configuration Archive	2330
Configuration Replace	2330
Configuration Rollback	2331
Configuration Guidelines	2331
Configuring the Configuration Archive	2332
Performing a Configuration Replacement or Rollback Operation	2333
Working with Software Images	2334
Information on Working with Software Images	2334
Image Location on the Switch	2335
File Format of Images on a Server or Cisco.com	2335
Copying Image Files Using TFTP	2336

Preparing to Download or Upload an Image File By Using TFTP	2336
Downloading an Image File By Using TFTP	2337
Uploading an Image File Using TFTP	2339
Copying Image Files Using FTP	2339
Preparing to Download or Upload an Image File By Using FTP	2340
Downloading an Image File By Using FTP	2341
Uploading an Image File By Using FTP	2343
Copying Image Files Using RCP	2344
Preparing to Download or Upload an Image File Using RCP	2344
Downloading an Image File using RCP	2345
Uploading an Image File using RCP	2347
Copying an Image File from One Stack Member to Another	2348



PART I

Interface and Hardware Component

- [Configuring Interface Characteristics, on page 1](#)
- [Configuring Auto-MDIX, on page 31](#)
- [Configuring Ethernet Management Port, on page 35](#)
- [Configuring LLDP, LLDP-MED, and Wired Location Service, on page 41](#)
- [Configuring System MTU, on page 59](#)
- [Configuring Internal Power Supplies, on page 63](#)
- [Configuring Boot Fast, on page 67](#)
- [Configuring PoE, on page 71](#)
- [Configuring 2-event Classification, on page 85](#)
- [Configuring EEE, on page 87](#)



CHAPTER

1

Configuring Interface Characteristics

- [Information About Configuring Interface Characteristics, on page 1](#)
- [How to Configure Interface Characteristics, on page 12](#)
- [Monitoring Interface Characteristics, on page 26](#)
- [Configuration Examples for Interface Characteristics, on page 27](#)
- [Additional References for the Interface Characteristics Feature, on page 30](#)
- [Feature History and Information for Configuring Interface Characteristics, on page 30](#)

Information About Configuring Interface Characteristics

Interface Types

This section describes the different types of interfaces supported by the device. The rest of the chapter describes configuration procedures for physical interface characteristics.



Note The stack ports on the rear of the stacking-capable devices are not Ethernet ports and cannot be configured.

Port-Based VLANs

A VLAN is a switched network that is logically segmented by function, team, or application, without regard to the physical location of the users. Packets received on a port are forwarded only to ports that belong to the same VLAN as the receiving port. Network devices in different VLANs cannot communicate with one another without a Layer 3 device to route traffic between the VLANs.

VLAN partitions provide hard firewalls for traffic in the VLAN, and each VLAN has its own MAC address table. A VLAN comes into existence when a local port is configured to be associated with the VLAN, when the VLAN Trunking Protocol (VTP) learns of its existence from a neighbor on a trunk, or when a user creates a VLAN. VLANs can be formed with ports across the stack.

To configure VLANs, use the **vlan *vlan-id*** global configuration command to enter VLAN configuration mode. The VLAN configurations for normal-range VLANs (VLAN IDs 1 to 1005) are saved in the VLAN database. If VTP is version 1 or 2, to configure extended-range VLANs (VLAN IDs 1006 to 4094), you must first set VTP mode to transparent. Extended-range VLANs created in transparent mode are not added to the VLAN

database but are saved in the device running configuration. With VTP version 3, you can create extended-range VLANs in client or server mode. These VLANs are saved in the VLAN database.

In a switch stack, the VLAN database is downloaded to all switches in a stack, and all switches in the stack build the same VLAN database. The running configuration and the saved configuration are the same for all switches in a stack.

Add ports to a VLAN by using the **switchport** interface configuration commands:

- Identify the interface.
- For a trunk port, set trunk characteristics, and, if desired, define the VLANs to which it can belong.
- For an access port, set and define the VLAN to which it belongs.
- For a tunnel port, set and define the VLAN ID for the customer-specific VLAN tag.

Switch Ports

Switch ports are Layer 2-only interfaces associated with a physical port. Switch ports belong to one or more VLANs. A switch port can be an access port or a trunk port. You can configure a port as an access port or trunk port or let the Dynamic Trunking Protocol (DTP) operate on a per-port basis to set the switchport mode by negotiating with the port on the other end of the link. switch ports are used for managing the physical interface and associated Layer 2 protocols and do not handle routing or bridging.

Configure switch ports by using the **switchport** interface configuration commands.

Access Ports

An access port belongs to and carries the traffic of only one VLAN (unless it is configured as a voice VLAN port). Traffic is received and sent in native formats with no VLAN tagging. Traffic arriving on an access port is assumed to belong to the VLAN assigned to the port. If an access port receives a tagged packet (Inter-Switch Link [ISL] or IEEE 802.1Q tagged), the packet is dropped, and the source address is not learned.

The types of access ports supported are:

- Static access ports are manually assigned to a VLAN (or through a RADIUS server for use with IEEE 802.1x).
- VLAN membership of dynamic access ports is learned through incoming packets. By default, a dynamic access port is not a member of any VLAN, and forwarding to and from the port is enabled only when the VLAN membership of the port is discovered. Dynamic access ports on the device are assigned to a VLAN by a VLAN Membership Policy Server (VMPS). The VMPS can be a Catalyst 6500 series switch; the device cannot be a VMPS server.

You can also configure an access port with an attached Cisco IP Phone to use one VLAN for voice traffic and another VLAN for data traffic from a device attached to the phone.

Trunk Ports

A trunk port carries the traffic of multiple VLANs and by default is a member of all VLANs in the VLAN database. These trunk port types are supported:

- In an ISL trunk port, all received packets are expected to be encapsulated with an ISL header, and all transmitted packets are sent with an ISL header. Native (non-tagged) frames received from an ISL trunk port are dropped.

- An IEEE 802.1Q trunk port supports simultaneous tagged and untagged traffic. An IEEE 802.1Q trunk port is assigned a default port VLAN ID (PVID), and all untagged traffic travels on the port default PVID. All untagged traffic and tagged traffic with a NULL VLAN ID are assumed to belong to the port default PVID. A packet with a VLAN ID equal to the outgoing port default PVID is sent untagged. All other traffic is sent with a VLAN tag.

Although by default, a trunk port is a member of every VLAN known to the VTP, you can limit VLAN membership by configuring an allowed list of VLANs for each trunk port. The list of allowed VLANs does not affect any other port but the associated trunk port. By default, all possible VLANs (VLAN ID 1 to 4094) are in the allowed list. A trunk port can become a member of a VLAN only if VTP knows of the VLAN and if the VLAN is in the enabled state. If VTP learns of a new, enabled VLAN and the VLAN is in the allowed list for a trunk port, the trunk port automatically becomes a member of that VLAN and traffic is forwarded to and from the trunk port for that VLAN. If VTP learns of a new, enabled VLAN that is not in the allowed list for a trunk port, the port does not become a member of the VLAN, and no traffic for the VLAN is forwarded to or from the port.

Tunnel Ports

Tunnel ports are used in IEEE 802.1Q tunneling to segregate the traffic of customers in a service-provider network from other customers who are using the same VLAN number. You configure an asymmetric link from a tunnel port on a service-provider edge switch to an IEEE 802.1Q trunk port on the customer switch. Packets entering the tunnel port on the edge switch, already IEEE 802.1Q-tagged with the customer VLANs, are encapsulated with another layer of an IEEE 802.1Q tag (called the metro tag), containing a VLAN ID unique in the service-provider network, for each customer. The double-tagged packets go through the service-provider network keeping the original customer VLANs separate from those of other customers. At the outbound interface, also a tunnel port, the metro tag is removed, and the original VLAN numbers from the customer network are retrieved.

Tunnel ports cannot be trunk ports or access ports and must belong to a VLAN unique to each customer.

Routed Ports

A routed port is a physical port that acts like a port on a router; it does not have to be connected to a router. A routed port is not associated with a particular VLAN, as is an access port. A routed port behaves like a regular router interface, except that it does not support VLAN subinterfaces. Routed ports can be configured with a Layer 3 routing protocol. A routed port is a Layer 3 interface only and does not support Layer 2 protocols, such as DTP and STP.

Configure routed ports by putting the interface into Layer 3 mode with the **no switchport** interface configuration command. Then assign an IP address to the port, enable routing, and assign routing protocol characteristics by using the **ip routing** and **router protocol** global configuration commands.



Note Entering a **no switchport** interface configuration command shuts down the interface and then re-enables it, which might generate messages on the device to which the interface is connected. When you put an interface that is in Layer 2 mode into Layer 3 mode, the previous configuration information related to the affected interface might be lost.

The number of routed ports that you can configure is not limited by software. However, the interrelationship between this number and the number of other features being configured might impact CPU performance because of hardware limitations.



Note The IP Base images supports static routing and the Routing Information Protocol (RIP). For full Layer 3 routing or for fallback bridging, you must enable the IP Services image on the standalone device, or the active device

Switch Virtual Interfaces

A switch virtual interface (SVI) represents a VLAN of switch ports as one interface to the routing or bridging function in the system. You can associate only one SVI with a VLAN. You configure an SVI for a VLAN only to route between VLANs or to provide IP host connectivity to the device. By default, an SVI is created for the default VLAN (VLAN 1) to permit remote device administration. Additional SVIs must be explicitly configured.



Note You cannot delete interface VLAN 1.

SVIs provide IP host connectivity only to the system. SVIs are created the first time that you enter the **vlan** interface configuration command for a VLAN interface. The VLAN corresponds to the VLAN tag associated with data frames on an ISL or IEEE 802.1Q encapsulated trunk or the VLAN ID configured for an access port. Configure a VLAN interface for each VLAN for which you want to route traffic, and assign it an IP address.

You can also use the interface range command to configure existing VLAN SVIs within the range. The commands entered under the interface range command are applied to all existing VLAN SVIs within the range. You can enter the command **interface range create vlan** *x - y* to create all VLANs in the specified range that do not already exist. When the VLAN interface is created, **interface range vlan** *id* can be used to configure the VLAN interface.

Although the switch stack or device supports a total of 1005 VLANs and SVIs, the interrelationship between the number of SVIs and routed ports and the number of other features being configured might impact CPU performance because of hardware limitations.

When you create an SVI, it does not become active until it is associated with a physical port.

SVI Autostate Exclude

The line state of an SVI with multiple ports on a VLAN is in the *up* state when it meets these conditions:

- The VLAN exists and is active in the VLAN database on the device
- The VLAN interface exists and is not administratively down.
- At least one Layer 2 (access or trunk) port exists, has a link in the *up* state on this VLAN, and is in the spanning-tree forwarding state on the VLAN.



Note The protocol link state for VLAN interfaces come up when the first switchport belonging to the corresponding VLAN link comes up and is in STP forwarding state.

The default action, when a VLAN has multiple ports, is that the SVI goes down when all ports in the VLAN go down. You can use the SVI autostate exclude feature to configure a port so that it is not included in the SVI line-state up-or-down calculation. For example, if the only active port on the VLAN is a monitoring port,

you might configure `autostate exclude` on that port so that the VLAN goes down when all other ports go down. When enabled on a port, **autostate exclude** applies to all VLANs that are enabled on that port.

The VLAN interface is brought up when one Layer 2 port in the VLAN has had time to converge (transition from STP listening-learning state to forwarding state). This prevents features such as routing protocols from using the VLAN interface as if it were fully operational and minimizes other problems.

EtherChannel Port Groups

EtherChannel port groups treat multiple switch ports as one switch port. These port groups act as a single logical port for high-bandwidth connections between devices or between devices and servers. An EtherChannel balances the traffic load across the links in the channel. If a link within the EtherChannel fails, traffic previously carried over the failed link changes to the remaining links. You can group multiple trunk ports into one logical trunk port, group multiple access ports into one logical access port, group multiple tunnel ports into one logical tunnel port, or group multiple routed ports into one logical routed port. Most protocols operate over either single ports or aggregated switch ports and do not recognize the physical ports within the port group. Exceptions are the DTP, the Cisco Discovery Protocol (CDP), and the Port Aggregation Protocol (PAgP), which operate only on physical ports.

When you configure an EtherChannel, you create a port-channel logical interface and assign an interface to the EtherChannel. For Layer 3 interfaces, you manually create the logical interface by using the **interface port-channel** global configuration command. Then you manually assign an interface to the EtherChannel by using the **channel-group** interface configuration command. For Layer 2 interfaces, use the **channel-group** interface configuration command to dynamically create the port-channel logical interface. This command binds the physical and logical ports together.

Power over Ethernet Ports

A PoE-capable switch port automatically supplies power to one of these connected devices if the device senses that there is no power on the circuit:

- a Cisco pre-standard powered device (such as a Cisco IP Phone or a Cisco Aironet Access Point)
- an IEEE 802.3af-compliant powered device

A powered device can receive redundant power when it is connected to a PoE switch port and to an AC power source. The device does not receive redundant power when it is only connected to the PoE port.

Using the Switch USB Ports

The device has three USB ports on the front panel — a USB mini-Type B console port and two USB Type A ports.

USB Mini-Type B Console Port

The device has the following console ports:

- USB mini-Type B console connection
- RJ-45 console port

Console output appears on devices connected to both ports, but console input is active on only one port at a time. By default, the USB connector takes precedence over the RJ-45 connector.



Note Windows PCs require a driver for the USB port. See the hardware installation guide for driver installation instructions.

Use the supplied USB Type A-to-USB mini-Type B cable to connect a PC or other device to the device. The connected device must include a terminal emulation application. When the device detects a valid USB connection to a powered-on device that supports host functionality (such as a PC), input from the RJ-45 console is immediately disabled, and input from the USB console is enabled. Removing the USB connection immediately reenables input from the RJ-45 console connection. An LED on the device shows which console connection is in use.

Console Port Change Logs

At software startup, a log shows whether the USB or the RJ-45 console is active. Each device in a stack issues this log. Every device always first displays the RJ-45 media type.

In the sample output, Device 1 has a connected USB console cable. Because the bootloader did not change to the USB console, the first log from Device 1 shows the RJ-45 console. A short time later, the console changes and the USB console log appears. Device 2 and Device 3 have connected RJ-45 console cables.

```
switch-stack-1
*Mar  1 00:01:00.171: %USB_CONSOLE-6-MEDIA_RJ45: Console media-type is RJ45.
*Mar  1 00:01:00.431: %USB_CONSOLE-6-MEDIA_USB: Console media-type is USB.
```

```
switch-stack-2
*Mar  1 00:01:09.835: %USB_CONSOLE-6-MEDIA_RJ45: Console media-type is RJ45.
```

```
switch-stack-3
*Mar  1 00:01:10.523: %USB_CONSOLE-6-MEDIA_RJ45: Console media-type is RJ45.
```

When the USB cable is removed or the PC de-activates the USB connection, the hardware automatically changes to the RJ-45 console interface:

```
switch-stack-1
Mar  1 00:20:48.635: %USB_CONSOLE-6-MEDIA_RJ45: Console media-type is RJ45.
```

You can configure the console type to always be RJ-45, and you can configure an inactivity timeout for the USB connector.

USB Type A Ports

The USB Type A ports provide access to external USB flash devices, also known as thumb drives or USB keys. The switch supports Cisco 64 MB, 256 MB, 512 MB, 1 GB, 4 GB, and 8 GB flash drives. You can use standard Cisco IOS command-line interface (CLI) commands to read, write, erase, and copy to or from the flash device. You can also configure the switch to boot from the USB flash drive.

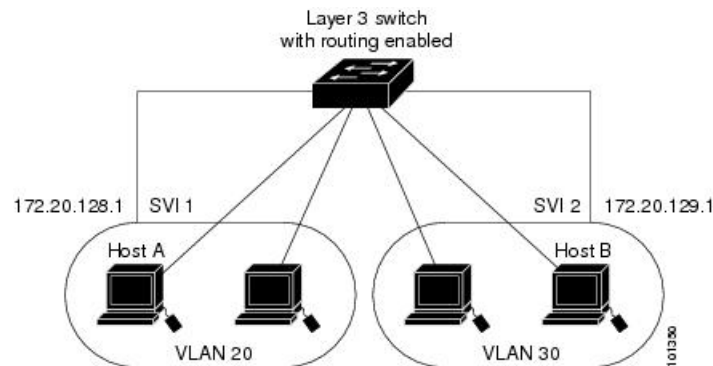
For information about configuring the switch to boot from a USB flash drive, refer to the *Catalyst 2960-XR Switch System Management Configuration Guide*.

For information about reading, writing, erasing, and copying files to or from the flash device, refer to the *Catalyst 2960-XR Switch Managing Cisco IOS Image Files Configuration Guide*.

Interface Connections

Devices within a single VLAN can communicate directly through any switch. Ports in different VLANs cannot exchange data without going through a routing device. With a standard Layer 2 device, ports in different VLANs have to exchange information through a router. By using the device with routing enabled, when you configure both VLAN 20 and VLAN 30 with an SVI to which an IP address is assigned, packets can be sent from Host A to Host B directly through the device with no need for an external router.

Figure 1: Connecting VLANs with the Switch



The IP Lite image IP Lite feature set supports static routing and RIP. The IP Base image supports only static routing. The routing function can be enabled on all SVIs and routed ports. The device routes only IP traffic. When IP routing protocol parameters and address configuration are added to an SVI or routed port, any IP traffic received from these ports is routed.



Note There is no limit to the number of static routes on devices running the IP Lite image.



Note Devices running the LAN Base image support configuring only 16 static routes on SVIs.

Interface Configuration Mode

The device supports these interface types:

- Physical ports—device ports and routed ports
- VLANs—switch virtual interfaces
- Port channels—EtherChannel interfaces

You can also configure a range of interfaces.

To configure a physical interface (port), specify the interface type, module number, and device port number, and enter interface configuration mode.

- Type—Gigabit Ethernet (gigabitethernet or gi) for 10/100/1000 Mb/s Ethernet ports, or small form-factor pluggable (SFP) module Gigabit Ethernet interfaces (gigabitethernet or gi).

- Stack member number—The number that identifies the switch within the stack. The range is 1 to 8 for a stack of Catalyst 2960-X switches, and 1 to 4 for a mixed stack of Catalyst 2960-XR, Catalyst 2960-X, and Catalyst 2960-S switches. The switch number is assigned the first time the switch initializes. The default switch number, before it is integrated into a switch stack, is 1. When a switch has been assigned a stack member number, it keeps that number until another is assigned to it.

You can use the switch port LEDs in Stack mode to identify the stack member number of a switch.

- Stack member number—The number that identifies the switch within the stack. The range is 1 to 8 for a stack of Catalyst 2960-XR switches. The switch number is assigned the first time the switch initializes. The default switch number, before it is integrated into a switch stack, is 1. When a switch has been assigned a stack member number, it keeps that number until another is assigned to it.

You can use the switch port LEDs in Stack mode to identify the stack member number of a switch.

- Module number—The module or slot number on the switch (always 0).
- Port number—The interface number on the switch. The 10/100/1000 port numbers always begin at 1, starting with the far left port when facing the front of the switch, for example, gigabitethernet1/0/1 or gigabitethernet1/0/8. For a switch with 10/100/1000 ports and SFP module ports, SFP module ports are numbered consecutively following the 10/100/1000 ports.

You can identify physical interfaces by physically checking the interface location on the switch. You can also use the **show** privileged EXEC commands to display information about a specific interface or all the interfaces on the switch. The remainder of this chapter primarily provides physical interface configuration procedures.

These are examples of how to identify interfaces on a stacking-capable switch:

- To configure 10/100/1000 port 4 on a standalone device, enter this command:

```
Device(config)# interface gigabitethernet1/0/4
```

- To configure 10/100/1000 port 4 on stack member 3, enter this command:

```
Device(config)# interface gigabitethernet1/0/4
```

Default Ethernet Interface Configuration

This table shows the Ethernet interface default configuration, including some features that apply only to Layer 2 interfaces.

Table 1: Default Layer 2 Ethernet Interface Configuration

Feature	Default Setting
Operating mode	Layer 2 or switching mode (switchport command).
Allowed VLAN range	VLANs 1– 4094.
Default VLAN (for access ports)	VLAN 1.
Native VLAN (for IEEE 802.1Q trunks)	VLAN 1.

Feature	Default Setting
802.1p priority-tagged traffic	Drop all packets tagged with VLAN 0.
VLAN trunking	Switchport mode dynamic auto (supports DTP).
Port enable state	All ports are enabled.
Port description	None defined.
Speed	Autonegotiate. (Not supported on the 10-Gigabit interfaces.)
Duplex mode	Autonegotiate. (Not supported on the 10-Gigabit interfaces.)
Flow control	Flow control is set to receive: off . It is always off for sent packets.
EtherChannel (PAgP)	Disabled on all Ethernet ports.
Port blocking (unknown multicast and unknown unicast traffic)	Disabled (not blocked).
Broadcast, multicast, and unicast storm control	Disabled.
Protected port	Disabled.
Port security	Disabled.
Port Fast	Disabled.
Auto-MDIX	Enabled. Note The device might not support a pre-standard powered device—such as Cisco IP phones and access points that do not fully support IEEE 802.3af—if that powered device is connected to the device through a crossover cable. This is regardless of whether auto-MIDX is enabled on the switch port.
Power over Ethernet (PoE)	Enabled (auto).
Keepalive messages	Disabled on SFP module ports; enabled on all other ports.

Interface Speed and Duplex Mode

Ethernet interfaces on the switch operate at 10, 100, 1000, or 10,000 Mb/s and in either full- or half-duplex mode. In full-duplex mode, two stations can send and receive traffic at the same time. Normally, 10-Mb/s ports operate in half-duplex mode, which means that stations can either receive or send traffic.

Switch modules include Gigabit Ethernet (10/100/1000-Mb/s) ports, 10-Gigabit Ethernet ports, and small form-factor pluggable (SFP) module slots supporting SFP modules.

Speed and Duplex Configuration Guidelines

When configuring an interface speed and duplex mode, note these guidelines:

- The 10-Gigabit Ethernet ports do not support the speed and duplex features. These ports operate only at 10,000 Mb/s and in full-duplex mode.
- Do not disable Auto-Negotiation on PoE switches.
- Gigabit Ethernet (10/100/1000-Mb/s) ports support all speed options and all duplex options (auto, half, and full). However, Gigabit Ethernet ports operating at 1000 Mb/s do not support half-duplex mode.
- For SFP module ports, the speed and duplex CLI options change depending on the SFP module type:
 - The 1000BASE-*x* (where *x* is -BX, -CWDM, -LX, -SX, and -ZX) SFP module ports support the **nonegotiate** keyword in the **speed** interface configuration command. Duplex options are not supported.
 - The 1000BASE-T SFP module ports support the same speed and duplex options as the 10/100/1000-Mb/s ports.
 -
- If both ends of the line support autonegotiation, we highly recommend the default setting of **auto** negotiation.
- If one interface supports autonegotiation and the other end does not, configure duplex and speed on both interfaces; do not use the **auto** setting on the supported side.
- When STP is enabled and a port is reconfigured, the device can take up to 30 seconds to check for loops. The port LED is amber while STP reconfigures.
- As best practice, we suggest configuring the speed and duplex options on a link to auto or to fixed on both the ends. If one side of the link is configured to auto and the other side is configured to fixed, the link will not be up and this is expected.



Caution

Changing the interface speed and duplex mode configuration might shut down and re-enable the interface during the reconfiguration.

IEEE 802.3x Flow Control

Flow control enables connected Ethernet ports to control traffic rates during congestion by allowing congested nodes to pause link operation at the other end. If one port experiences congestion and cannot receive any more traffic, it notifies the other port by sending a pause frame to stop sending until the condition clears. Upon receipt of a pause frame, the sending device stops sending any data packets, which prevents any loss of data packets during the congestion period.



Note

The switch ports can receive, but not send, pause frames.

You use the **flowcontrol** interface configuration command to set the interface's ability to **receive** pause frames to **on**, **off**, or **desired**. The default state is **off**.

When set to **desired**, an interface can operate with an attached device that is required to send flow-control packets or with an attached device that is not required to but can send flow-control packets.

These rules apply to flow control settings on the device:

- **receive on (or desired)**: The port cannot send pause frames but can operate with an attached device that is required to or can send pause frames; the port can receive pause frames.
- **receive off**: Flow control does not operate in either direction. In case of congestion, no indication is given to the link partner, and no pause frames are sent or received by either device.



Note For details on the command settings and the resulting flow control resolution on local and remote ports, see the **flowcontrol** interface configuration command in the command reference for this release.

Layer 3 Interfaces

The device supports these types of Layer 3 interfaces:

- **SVIs**: You should configure SVIs for any VLANs for which you want to route traffic. SVIs are created when you enter a VLAN ID following the **interface vlan** global configuration command. To delete an SVI, use the **no interface vlan** global configuration command. You cannot delete interface VLAN 1.



Note When you create an SVI, it does not become active until it is associated with a physical port.

When configuring SVIs, you can also configure SVI autostate exclude on a port in the SVI to exclude that port from being included in determining SVI line-state status.

- **Routed ports**: Routed ports are physical ports configured to be in Layer 3 mode by using the **no switchport** interface configuration command.
- **Layer 3 EtherChannel ports**: EtherChannel interfaces made up of routed ports.

A Layer 3 device can have an IP address assigned to each routed port and SVI.

There is no defined limit to the number of SVIs and routed ports that can be configured in a device or in a device stack. However, the interrelationship between the number of SVIs and routed ports and the number of other features being configured might have an impact on CPU usage because of hardware limitations. If the device is using its maximum hardware resources, attempts to create a routed port or SVI have these results:

- If you try to create a new routed port, the device generates a message that there are not enough resources to convert the interface to a routed port, and the interface remains as a switchport.
- If you try to create an extended-range VLAN, an error message is generated, and the extended-range VLAN is rejected.

- If the device is notified by VLAN Trunking Protocol (VTP) of a new VLAN, it sends a message that there are not enough hardware resources available and shuts down the VLAN. The output of the **show vlan** user EXEC command shows the VLAN in a suspended state.
- If the device attempts to boot up with a configuration that has more VLANs and routed ports than hardware can support, the VLANs are created, but the routed ports are shut down, and the device sends a message that this was due to insufficient hardware resources.



Note All Layer 3 interfaces require an IP address to route traffic. This procedure shows how to configure an interface as a Layer 3 interface and how to assign an IP address to an interface:

If the physical port is in Layer 2 mode (the default), you must enter the **no switchport** interface configuration command to put the interface into Layer 3 mode. Entering a **no switchport** command disables and then re-enables the interface, which might generate messages on the device to which the interface is connected. Furthermore, when you put an interface that is in Layer 2 mode into Layer 3 mode, the previous configuration information related to the affected interface might be lost, and the interface is returned to its default configuration

How to Configure Interface Characteristics

Configuring Interfaces

These general instructions apply to all interface configuration processes.

Procedure

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	interface Example: Device(config)# interface gigabitethernet 1/0/1	Identifies the interface type, the device number (only on stacking-capable switches), and the number of the connector.

	Command or Action	Purpose
	Device (config-if) #	Note You do not need to add a space between the interface type and the interface number. For example, in the preceding line, you can specify either gigabitethernet 1/0/1 , gigabitethernet1/0/1 , gi 1/0/1 , or gi1/0/1 .
Step 4	Follow each interface command with the interface configuration commands that the interface requires.	Defines the protocols and applications that will run on the interface. The commands are collected and applied to the interface when you enter another interface command or enter end to return to privileged EXEC mode.
Step 5	interface range or interface range macro	(Optional) Configures a range of interfaces. Note Interfaces configured in a range must be the same type and must be configured with the same feature options.
Step 6	show interfaces	Displays a list of all interfaces on or configured for the switch. A report is provided for each interface that the device supports or for the specified interface.

Adding a Description for an Interface

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **description** *string*
5. **end**
6. **show interfaces** *interface-id* **description**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.

	Command or Action	Purpose
Step 3	interface <i>interface-id</i> Example: Device(config) # interface gigabitethernet 1/0/2	Specifies the interface for which you are adding a description, and enter interface configuration mode.
Step 4	description <i>string</i> Example: Device(config-if) # description Connects to Marketing	Adds a description (up to 240 characters) for an interface.
Step 5	end Example: Device(config-if) # end	Returns to privileged EXEC mode.
Step 6	show interfaces <i>interface-id</i> description	Verifies your entry.
Step 7	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Configuring a Range of Interfaces

To configure multiple interfaces with the same configuration parameters, use the **interface range** global configuration command. When you enter the interface-range configuration mode, all command parameters that you enter are attributed to all interfaces within that range until you exit this mode.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface range** {*port-range* | **macro** *macro_name*}
4. **end**
5. **show interfaces** [*interface-id*]
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example:	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.

	Command or Action	Purpose
	Device> <code>enable</code>	
Step 2	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode.
Step 3	interface range { <i>port-range</i> macro <i>macro_name</i> } Example: Device(config)# <code>interface range macro</code>	Specifies the range of interfaces (VLANs or physical ports) to be configured, and enter interface-range configuration mode. <ul style="list-style-type: none"> You can use the interface range command to configure up to five port ranges or a previously defined macro. The macro variable is explained in the section on <i>Configuring and Using Interface Range Macros</i>. In a comma-separated <i>port-range</i>, you must enter the interface type for each entry and enter spaces before and after the comma. In a hyphen-separated <i>port-range</i>, you do not need to re-enter the interface type, but you must enter a space before the hyphen. <p>Note Use the normal configuration commands to apply the configuration parameters to all interfaces in the range. Each command is executed as it is entered.</p>
Step 4	end Example: Device(config)# <code>end</code>	Returns to privileged EXEC mode.
Step 5	show interfaces [<i>interface-id</i>] Example: Device# <code>show interfaces</code>	Verifies the configuration of the interfaces in the range.
Step 6	copy running-config startup-config Example: Device# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Configuring and Using Interface Range Macros

You can create an interface range macro to automatically select a range of interfaces for configuration. Before you can use the **macro** keyword in the **interface range macro** global configuration command string, you must use the **define interface-range** global configuration command to define the macro.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **define interface-range** *macro_name* *interface-range*
4. **interface range macro** *macro_name*
5. **end**
6. **show running-config** | **include define**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	define interface-range <i>macro_name</i> <i>interface-range</i> Example: Device(config)# define interface-range enet_list gigabitethernet 1/0/1 - 2	Defines the interface-range macro, and save it in NVRAM. <ul style="list-style-type: none"> • The <i>macro_name</i> is a 32-character maximum character string. • A macro can contain up to five comma-separated interface ranges. • Each <i>interface-range</i> must consist of the same port type. <p>Note Before you can use the macro keyword in the interface range macro global configuration command string, you must use the define interface-range global configuration command to define the macro.</p>
Step 4	interface range macro <i>macro_name</i> Example:	Selects the interface range to be configured using the values saved in the interface-range macro called <i>macro_name</i> .

	Command or Action	Purpose
	Device(config)# <code>interface range macro enet_list</code>	You can now use the normal configuration commands to apply the configuration to all interfaces in the defined macro.
Step 5	end Example: Device(config)# <code>end</code>	Returns to privileged EXEC mode.
Step 6	show running-config include define Example: Device# <code>show running-config include define</code>	Shows the defined interface range macro configuration.
Step 7	copy running-config startup-config Example: Device# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Configuring Ethernet Interfaces

Setting the Interface Speed and Duplex Parameters

SUMMARY STEPS

1. `enable`
2. `configure terminal`
3. `interface interface-id`
4. `speed {10 | 100 | 1000 | 2500 | 5000 | 10000 | auto [10 | 100 | 1000 | 2500 | 5000 | 10000] | nonegotiate}`
5. `duplex {auto | full | half}`
6. `end`
7. `show interfaces interface-id`
8. `copy running-config startup-config`

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> <code>enable</code>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.

	Command or Action	Purpose
Step 2	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode.
Step 3	interface <i>interface-id</i> Example: Device(config)# <code>interface gigabitethernet 1/0/3</code>	Specifies the physical interface to be configured, and enter interface configuration mode.
Step 4	speed {10 100 1000 2500 5000 10000 auto [10 100 1000 2500 5000 10000] nonegotiate} Example: Device(config-if)# <code>speed 10</code>	Enter the appropriate speed parameter for the interface: <ul style="list-style-type: none"> • Enter 10, 100, 1000, 2500, 5000, or 10000 to set a specific speed for the interface. • Enter auto to enable the interface to autonegotiate speed with the connected device. If you specify a speed and also set the auto keyword, the port autonegotiates only at the specified speeds. • The nonegotiate keyword is available only for SFP module ports. SFP module ports operate only at 1000 Mb/s but can be configured to not negotiate if connected to a device that does not support autonegotiation.
Step 5	duplex {auto full half} Example: Device(config-if)# <code>duplex half</code>	This command is not available on a 10-Gigabit Ethernet interface. Enter the duplex parameter for the interface. Enable half-duplex mode (for interfaces operating only at 10 or 100 Mb/s). You cannot configure half-duplex mode for interfaces operating at 1000 Mb/s. You can configure the duplex setting when the speed is set to auto .
Step 6	end Example: Device(config-if)# <code>end</code>	Returns to privileged EXEC mode.
Step 7	show interfaces <i>interface-id</i> Example: Device# <code>show interfaces gigabitethernet 1/0/3</code>	Displays the interface speed and duplex mode configuration.

	Command or Action	Purpose
Step 8	copy running-config startup-config Example: <pre>Device# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Configuring IEEE 802.3x Flow Control

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **flowcontrol** {receive} {on | off | desired}
4. **end**
5. **show interfaces** *interface-id*
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>Device# configure terminal</pre>	Enters global configuration mode
Step 2	interface <i>interface-id</i> Example: <pre>Device(config)# interface gigabitethernet 1/0/1</pre>	Specifies the physical interface to be configured, and enter interface configuration mode.
Step 3	flowcontrol {receive} {on off desired} Example: <pre>Device(config-if)# flowcontrol receive on</pre>	Configures the flow control mode for the port.
Step 4	end Example: <pre>Device(config-if)# end</pre>	Returns to privileged EXEC mode.
Step 5	show interfaces <i>interface-id</i> Example:	Verifies the interface flow control settings.

	Command or Action	Purpose
	Device# <code>show interfaces gigabitethernet 1/0/1</code>	
Step 6	copy running-config startup-config Example: Device# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Configuring Layer 3 Interfaces

SUMMARY STEPS

1. `enable`
2. `configure terminal`
3. `interface {gigabitethernet interface-id} | {vlan vlan-id} | {port-channel port-channel-number}`
4. `no switchport`
5. `ip address ip_address subnet_mask`
6. `no shutdown`
7. `end`
8. `show interfaces [interface-id]`
9. `copy running-config startup-config`

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> <code>enable</code>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode.
Step 3	interface {gigabitethernet <i>interface-id</i>} {vlan <i>vlan-id</i>} {port-channel <i>port-channel-number</i>} Example: Device(config)# <code>interface gigabitethernet1/0/2</code>	Specifies the interface to be configured as a Layer 3 interface, and enter interface configuration mode.

	Command or Action	Purpose
Step 4	no switchport Example: Device(config-if)# no switchport	For physical ports only, enters Layer 3 mode.
Step 5	ip address ip_address subnet_mask Example: Device(config-if)# ip address 192.20.135.21 255.255.255.0	Configures the IP address and IP subnet.
Step 6	no shutdown Example: Device(config-if)# no shutdown	Enables the interface.
Step 7	end Example: Device(config-if)# end	Returns to privileged EXEC mode.
Step 8	show interfaces [interface-id]	Verifies the configuration.
Step 9	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Configuring SVI Autostate Exclude

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface interface-id**
4. **switchport autostate exclude**
5. **end**
6. **show running config interface interface-id**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none">• Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	interface <i>interface-id</i> Example: Device(config)# interface gigabitethernet1/0/2	Specifies a Layer 2 interface (physical port or port channel), and enter interface configuration mode.
Step 4	switchport autostate exclude Example: Device(config-if)# switchport autostate exclude	Excludes the access or trunk port when defining the status of an SVI line state (up or down)
Step 5	end Example: Device(config-if)# end	Returns to privileged EXEC mode.
Step 6	show running config interface <i>interface-id</i>	(Optional) Shows the running configuration. Verifies the configuration.
Step 7	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Shutting Down and Restarting the Interface

Shutting down an interface disables all functions on the specified interface and marks the interface as unavailable on all monitoring command displays. This information is communicated to other network servers through all dynamic routing protocols. The interface is not mentioned in any routing updates.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** {vlan *vlan-id*} | { **gigabitethernet***interface-id*} | {**port-channel** *port-channel-number*}
4. **shutdown**
5. **no shutdown**
6. **end**
7. **show running-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	interface {vlan <i>vlan-id</i> } { gigabitethernet <i>interface-id</i> } { port-channel <i>port-channel-number</i> } Example: Device(config)# interface gigabitethernet 1/0/2	Selects the interface to be configured.
Step 4	shutdown Example: Device(config-if)# shutdown	Shuts down an interface.
Step 5	no shutdown Example: Device(config-if)# no shutdown	Restarts an interface.
Step 6	end Example: Device(config-if)# end	Returns to privileged EXEC mode.

	Command or Action	Purpose
Step 7	show running-config Example: Device# <code>show running-config</code>	Verifies your entries.

Configuring the Console Media Type

Follow these steps to set the console media type to RJ-45. If you configure the console as RJ-45, USB console operation is disabled, and input comes only through the RJ-45 connector.

This configuration applies to all switches in a stack.

SUMMARY STEPS

1. `enable`
2. `configure terminal`
3. `line console 0`
4. `media-type rj45`
5. `end`
6. `copy running-config startup-config`

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> <code>enable</code>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode.
Step 3	line console 0 Example: Device(config)# <code>line console 0</code>	Configures the console and enters line configuration mode.

	Command or Action	Purpose
Step 4	media-type rj45 Example: <pre>Device(config-line)# media-type rj45</pre>	Configures the console media type to be only RJ-45 port. If you do not enter this command and both types are connected, the USB port is used by default.
Step 5	end Example: <pre>Device(config)# end</pre>	Returns to privileged EXEC mode.
Step 6	copy running-config startup-config Example: <pre>Device# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Configuring the USB Inactivity Timeout

The configurable inactivity timeout reactivates the RJ-45 console port if the USB console port is activated but no input activity occurs on it for a specified time period. When the USB console port is deactivated due to a timeout, you can restore its operation by disconnecting and reconnecting the USB cable.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **line console 0**
4. **usb-inactivity-timeout *timeout-minutes***
5. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Device> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: <pre>Device# configure terminal</pre>	Enters global configuration mode.

	Command or Action	Purpose
Step 3	line console 0 Example: Device(config)# line console 0	Configures the console and enters line configuration mode.
Step 4	usb-inactivity-timeout <i>timeout-minutes</i> Example: Device(config-line)# usb-inactivity-timeout 30	Specify an inactivity timeout for the console port. The range is 1 to 240 minutes. The default is to have no timeout configured.
Step 5	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Monitoring Interface Characteristics

Monitoring Interface Status

Commands entered at the privileged EXEC prompt display information about the interface, including the versions of the software and the hardware, the configuration, and statistics about the interfaces.

Table 2: Show Commands for Interfaces

Command	Purpose
show interfaces <i>interface-id</i> status [err-disabled]	Displays interface status or a list of interfaces in the error-disabled state.
show interfaces [<i>interface-id</i>] switchport	Displays administrative and operational status of switching (nonrouting) ports. You can use this command to find out if a port is in routing or in switching mode.
show interfaces [<i>interface-id</i>] description	Displays the description configured on an interface or all interfaces and the interface status.
show ip interface [<i>interface-id</i>]	Displays the usability status of all interfaces configured for IP routing or the specified interface.
show interface [<i>interface-id</i>] stats	Displays the input and output packets by the switching path for the interface.
show interfaces <i>interface-id</i>	(Optional) Displays speed and duplex on the interface.

Command	Purpose
show interfaces transceiver dom-supported-list	(Optional) Displays Digital Optical Monitoring (DOM) status on the connect SFP modules.
show interfaces transceiver properties	(Optional) Displays temperature, voltage, or amount of current on the interface.
show interfaces [<i>interface-id</i>] [{ transceiver properties detail }] <i>module number</i>	Displays physical and operational status about an SFP module.
show running-config interface [<i>interface-id</i>]	Displays the running configuration in RAM for the interface.
show version	Displays the hardware configuration, software version, the names and sources of configuration files, and the boot images.
show controllers ethernet-controller <i>interface-id</i> phy	Displays the operational state of the auto-MDIX feature on the interface.

Clearing and Resetting Interfaces and Counters

Table 3: Clear Commands for Interfaces

Command	Purpose
clear counters [<i>interface-id</i>]	Clears interface counters.
clear interface <i>interface-id</i>	Resets the hardware logic on an interface.
clear line [<i>number</i> console 0 vty number]	Resets the hardware logic on an asynchronous serial line.



Note The **clear counters** privileged EXEC command does not clear counters retrieved by using Simple Network Management Protocol (SNMP), but only those seen with the **show interface** privileged EXEC command.

Configuration Examples for Interface Characteristics

Configuring a Range of Interfaces: Examples

This example shows how to use the **interface range** global configuration command to set the speed to 100 Mb/s on ports 1 to 4 on switch 1:

```
Device# configure terminal
Device(config)# interface range gigabitethernet 1/0/1 - 4
Device(config-if-range)# speed 100
```

This example shows how to use a comma to add different interface type strings to the range to enable Gigabit Ethernet ports 1 to 3 and 10-Gigabit Ethernet ports 1 and 2 to receive flow-control pause frames:

```
Device# configure terminal
Device(config)# interface range gigabitethernet1/0/1 - 3 , tengigabitethernet1/1/1 - 2
Device(config-if-range)# flowcontrol receive on
```

If you enter multiple configuration commands while you are in interface-range mode, each command is executed as it is entered. The commands are not batched and executed after you exit interface-range mode. If you exit interface-range configuration mode while the commands are being executed, some commands might not be executed on all interfaces in the range. Wait until the command prompt reappears before exiting interface-range configuration mode.

Configuring and Using Interface Range Macros: Examples

This example shows how to define an interface-range named *enet_list* to include ports 1 and 2 on switch 1 and to verify the macro configuration:

```
Device# configure terminal
Device(config)# define interface-range enet_list gigabitethernet 1/1/1 - 2
Device(config)# end
Device# show running-config | include define
define interface-range enet_list gigabitethernet 1/1/1 - 2
```

This example shows how to create a multiple-interface macro named *macro1*:

```
Device# configure terminal
Device(config)# define interface-range macro1 gigabitethernet1/1/1 - 2, gigabitethernet1/1/5
- 7, tengigabitethernet1/1/1 -2
Device(config)# end
```

This example shows how to enter interface-range configuration mode for the interface-range macro *enet_list*:

```
Device# configure terminal
Device(config)# interface range macro enet_list
Device(config-if-range)#
```

This example shows how to delete the interface-range macro *enet_list* and to verify that it was deleted.

```
Device# configure terminal
Device(config)# no define interface-range enet_list
Device(config)# end
Device# show run | include define
Device#
```

Setting Interface Speed and Duplex Mode: Example

This example shows how to set the interface speed to 100 Mb/s and the duplex mode to half on a 10/100/1000 Mb/s port:

```
Device# configure terminal
```

```
Device(config)# interface gigabitethernet 1/0/3
Device(config-if)# speed 10
Device(config-if)# duplex half
```

This example shows how to set the interface speed to 100 Mb/s on a 10/100/1000 Mb/s port:

```
Device# configure terminal
Device(config)# interface gigabitethernet 1/0/2
Device(config-if)# speed 100
```

Configuring Layer 3 Interfaces: Example

```
Device# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Device(config)# interface gigabitethernet1/0/2
Device(config-if)# no switchport
Device(config-if)# ip address 192.20.135.21 255.255.255.0
Device(config-if)# no shutdown
```

Configuring the Console Media Type: Example

This example disables the USB console media type and enables the RJ-45 console media type.

```
Device# configure terminal
Device(config)# line console 0
Device(config-line)# media-type rj45
```

This example reverses the previous configuration and immediately activates any USB console that is connected.

```
Device# configure terminal
Device(config)# line console 0
Device(config-line)# no media-type rj45
```

Configuring the USB Inactivity Timeout: Example

This example configures the inactivity timeout to 30 minutes:

```
Device# configure terminal
Device(config)# line console 0
Device(config-line)# usb-inactivity-timeout 30
```

To disable the configuration, use these commands:

```
Device# configure terminal
Device(config)# line console 0
Device(config-line)# no usb-inactivity-timeout
```

If there is no (input) activity on a USB console port for the configured number of minutes, the inactivity timeout setting applies to the RJ-45 port, and a log shows this occurrence:

```
*Mar 1 00:47:25.625: %USB_CONSOLE-6-INACTIVITY_DISABLE: Console media-type USB disabled
due to inactivity, media-type reverted to RJ45.
```

At this point, the only way to reactivate the USB console port is to disconnect and reconnect the cable.

When the USB cable on the switch has been disconnected and reconnected, a log similar to this appears:

```
*Mar 1 00:48:28.640: %USB_CONSOLE-6-MEDIA_USB: Console media-type is USB.
```

Additional References for the Interface Characteristics Feature

Standards and RFCs

Standard/RFC	Title
None	--

MIBs

MIB	MIBs Link
All the supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
<p>The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies.</p> <p>To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds.</p> <p>Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.</p>	http://www.cisco.com/support

Feature History and Information for Configuring Interface Characteristics

Release	Modification
Cisco IOS Release 15.0(2)EX1	This feature was introduced.



CHAPTER 2

Configuring Auto-MDIX

- [Prerequisites for Auto-MDIX, on page 31](#)
- [Restrictions for Auto-MDIX, on page 31](#)
- [Information About Configuring Auto-MDIX, on page 32](#)
- [How to Configure Auto-MDIX, on page 32](#)
- [Example for Configuring Auto-MDIX, on page 33](#)
- [Additional References, on page 34](#)
- [Feature History and Information for Auto-MDIX, on page 34](#)

Prerequisites for Auto-MDIX

To configure Layer 2 parameters, if the interface is in Layer 3 mode, you must enter the **switchport** interface configuration command without any parameters to put the interface into Layer 2 mode. This shuts down the interface and then re-enables it, which might generate messages on the device to which the interface is connected. When you put an interface that is in Layer 3 mode into Layer 2 mode, the previous configuration information related to the affected interface might be lost, and the interface is returned to its default configuration.

Automatic medium-dependent interface crossover (auto-MDIX) is enabled by default.

Auto-MDIX is supported on all 10/100/1000-Mb/s and on 10/100/1000BASE-TX small form-factor pluggable (SFP)-module interfaces. It is not supported on 1000BASE-SX or -LX SFP module interfaces.

Restrictions for Auto-MDIX

The device might not support a pre-standard powered device—such as Cisco IP phones and access points that do not fully support IEEE 802.3af—if that powered device is connected to the device through a crossover cable. This is regardless of whether auto-MDIX is enabled on the switch port.

Information About Configuring Auto-MDIX

Auto-MDIX on an Interface

When automatic medium-dependent interface crossover (auto-MDIX) is enabled on an interface, the interface automatically detects the required cable connection type (straight through or crossover) and configures the connection appropriately. When connecting devices without the auto-MDIX feature, you must use straight-through cables to connect to devices such as servers, workstations, or routers and crossover cables to connect to other devices or repeaters. With auto-MDIX enabled, you can use either type of cable to connect to other devices, and the interface automatically corrects for any incorrect cabling. For more information about cabling requirements, see the hardware installation guide.

This table shows the link states that result from auto-MDIX settings and correct and incorrect cabling.

Table 4: Link Conditions and Auto-MDIX Settings

Local Side Auto-MDIX	Remote Side Auto-MDIX	With Correct Cabling	With Incorrect Cabling
On	On	Link up	Link up
On	Off	Link up	Link up
Off	On	Link up	Link up
Off	Off	Link up	Link down

How to Configure Auto-MDIX

Configuring Auto-MDIX on an Interface

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **speed auto**
5. **duplex auto**
6. **end**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example:	Enables privileged EXEC mode. • Enter your password if prompted.

	Command or Action	Purpose
	Device> enable	
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode
Step 3	interface interface-id Example: Device(config)# interface gigabitethernet 1/0/1	Specifies the physical interface to be configured, and enter interface configuration mode.
Step 4	speed auto Example: Device(config-if)# speed auto	Configures the interface to autonegotiate speed with the connected device.
Step 5	duplex auto Example: Device(config-if)# duplex auto	Configures the interface to autonegotiate duplex mode with the connected device.
Step 6	end Example: Device(config-if)# end	Returns to privileged EXEC mode.
Step 7	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Example for Configuring Auto-MDIX

This example shows how to enable auto-MDIX on a port:

```
Device# configure terminal
Device(config)# interface gigabitethernet 1/0/1
Device(config-if)# speed auto
Device(config-if)# duplex auto
Device(config-if)# mdix auto
```

```
Device(config-if)# end
```

Additional References

MIBs

MIB	MIBs Link
All the supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
<p>The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies.</p> <p>To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds.</p> <p>Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.</p>	http://www.cisco.com/support

Feature History and Information for Auto-MDIX

Release	Modification
Cisco IOS Release 15.0(2)EX1	This feature was introduced.



CHAPTER 3

Configuring Ethernet Management Port

- [Finding Feature Information, on page 35](#)
- [Prerequisites for Ethernet Management Ports, on page 35](#)
- [Information About the Ethernet Management Port, on page 35](#)
- [How to Configure the Ethernet Management Port, on page 38](#)
- [Additional References for Ethernet Management Ports, on page 39](#)
- [Feature History and Information for Ethernet Management Ports, on page 39](#)

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to <http://www.cisco.com/go/cfn>. An account on Cisco.com is not required.

Prerequisites for Ethernet Management Ports

When connecting a PC to the Ethernet management port, you must first assign an IP address.

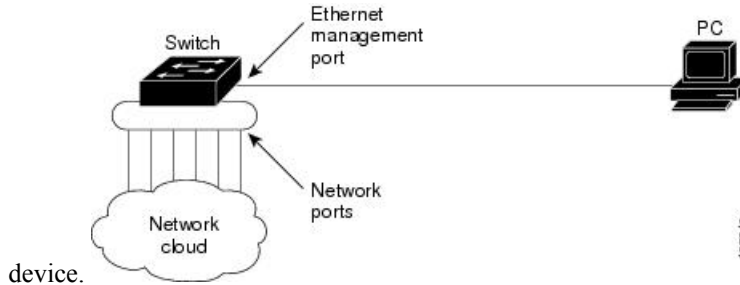
Information About the Ethernet Management Port

The Ethernet management port, also referred to as the *Fa0* or *fastethernet0* port, is a Layer 3 host port to which you can connect a PC. You can use the Ethernet management port instead of the device console port for network management. When managing a device stack, connect the PC to the Ethernet management port on a stack member.

Ethernet Management Port Direct Connection to a Device

Figure 2: Connecting a Switch to a PC

This figure displays how to connect the Ethernet management port to the PC for a device or a standalone



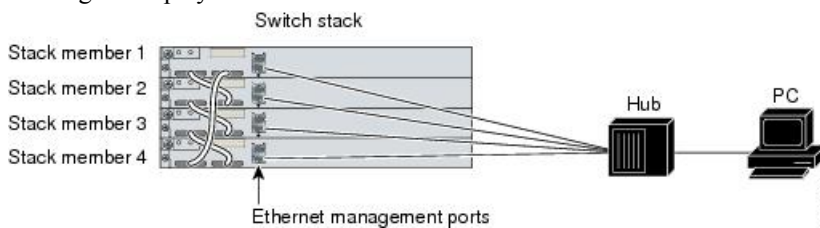
device.

Ethernet Management Port Connection to Stack Devices using a Hub

In a stack with only stack devices, all the Ethernet management ports on the stack members are connected to a hub to which the PC is connected. The active link is from the Ethernet management port on the active switchstack's active switch through the hub, to the PC. If the active device fails and a new active device is elected, the active link is now from the Ethernet management port on the new active device to the PC.

Figure 3: Connecting a Device Stack to a PC

This figure displays how a PC uses a hub to connect to a device stack.

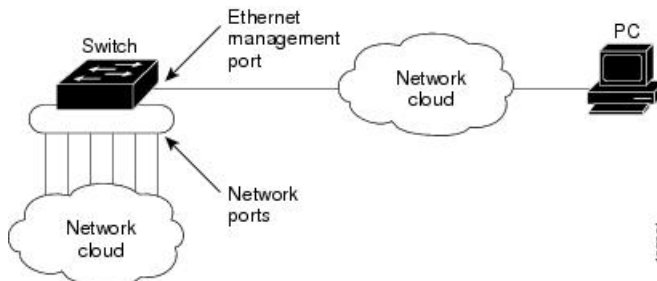


Ethernet Management Port and Routing

By default, the Ethernet management port is enabled. The device cannot route packets from the Ethernet management port to a network port, and the reverse. Even though the Ethernet management port does not support routing, you may need to enable routing protocols on the port.

Figure 4: Network Example with Routing Protocols Enabled

Enable routing protocols on the Ethernet management port when the PC is multiple hops away from the device and the packets must pass through multiple Layer 3 devices to reach the PC.



In the above figure, if the Ethernet management port and the network ports are associated with the same routing process, the routes are propagated as follows:

- The routes from the Ethernet management port are propagated through the network ports to the network.
- The routes from the network ports are propagated through the Ethernet management port to the network.

Because routing is not supported between the Ethernet management port and the network ports, traffic between these ports cannot be sent or received. If this happens, data packet loops occur between the ports, which disrupt the device and network operation. To prevent the loops, configure route filters to avoid routes between the Ethernet management port and the network ports.

Supported Features on the Ethernet Management Port

The Ethernet management port supports these features:

- Express Setup (only in switch stacks)
- Network Assistant
- Telnet with passwords
- TFTP
- Secure Shell (SSH)
- DHCP-based autoconfiguration
- SNMP (only the ENTITY-MIB and the IF-MIB)
- IP ping
- Interface features
 - Speed—10 Mb/s, 100 Mb/s, and autonegotiation
 - Duplex mode—Full, half, and autonegotiation
 - Loopback detection
- Cisco Discovery Protocol (CDP)
- DHCP relay agent
- IPv4 and IPv6 access control lists (ACLs)
- Routing protocols



Caution

Before enabling a feature on the Ethernet management port, make sure that the feature is supported. If you try to configure an unsupported feature on the Ethernet Management port, the feature might not work properly, and the device might fail.

How to Configure the Ethernet Management Port

Disabling and Enabling the Ethernet Management Port

SUMMARY STEPS

1. `configure terminal`
2. `interface fastethernet0`
3. `shutdown`
4. `no shutdown`
5. `exit`
6. `show interfaces fastethernet0`

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode.
Step 2	interface fastethernet0 Example: Device(config)# <code>interface fastethernet0</code>	Specifies the Ethernet management port in the CLI.
Step 3	shutdown Example: Device(config-if)# <code>shutdown</code>	Disables the Ethernet management port.
Step 4	no shutdown Example: Device(config-if)# <code>no shutdown</code>	Enables the Ethernet management port.
Step 5	exit Example: Device(config-if)# <code>exit</code>	Exits interface configuration mode.
Step 6	show interfaces fastethernet0 Example: Device# <code>show interfaces fastethernet0</code>	Displays the link status. To find out the link status to the PC, you can monitor the LED for the Ethernet management port. The LED is green (on) when the link is active, and the LED is off when the link is down. The LED is amber when there is a POST failure.

What to do next

Proceed to manage or configure your switch using the Ethernet management port. Refer to the *Catalyst 2960-XR Switch Network Management Configuration Guide*.

Additional References for Ethernet Management Ports

Related Documents

Related Topic	Document Title
Bootloader configuration	<i>Catalyst 2960-XR Switch System Management Configuration Guide</i>
Bootloader commands	<i>Catalyst 2960-XR Switch System Management Command Reference</i>

MIBs

MIB	MIBs Link
All the supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
<p>The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies.</p> <p>To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds.</p> <p>Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.</p>	http://www.cisco.com/support

Feature History and Information for Ethernet Management Ports

Release	Modification
Cisco IOS Release 15.0(2)EX1	This feature was introduced.



CHAPTER 4

Configuring LLDP, LLDP-MED, and Wired Location Service

- [Information About LLDP, LLDP-MED, and Wired Location Service, on page 41](#)
- [How to Configure LLDP, LLDP-MED, and Wired Location Service, on page 45](#)
- [Configuration Examples for LLDP, LLDP-MED, and Wired Location Service, on page 56](#)
- [Monitoring and Maintaining LLDP, LLDP-MED, and Wired Location Service, on page 57](#)
- [Additional References for LLDP, LLDP-MED, and Wired Location Service, on page 58](#)
- [Feature Information for LLDP, LLDP-MED, and Wired Location Service, on page 58](#)

Information About LLDP, LLDP-MED, and Wired Location Service

LLDP

The Cisco Discovery Protocol (CDP) is a device discovery protocol that runs over Layer 2 (the data link layer) on all Cisco-manufactured devices (routers, bridges, access servers, switches, and controllers). CDP allows network management applications to automatically discover and learn about other Cisco devices connected to the network.

To support non-Cisco devices and to allow for interoperability between other devices, the device supports the IEEE 802.1AB Link Layer Discovery Protocol (LLDP). LLDP is a neighbor discovery protocol that is used for network devices to advertise information about themselves to other devices on the network. This protocol runs over the data-link layer, which allows two systems running different network layer protocols to learn about each other.

LLDP Supported TLVs

LLDP supports a set of attributes that it uses to discover neighbor devices. These attributes contain type, length, and value descriptions and are referred to as TLVs. LLDP supported devices can use TLVs to receive and send information to their neighbors. This protocol can advertise details such as configuration information, device capabilities, and device identity.

The switch supports these basic management TLVs. These are mandatory LLDP TLVs.

- Port description TLV
- System name TLV

- System description TLV
- System capabilities TLV
- Management address TLV

These organizationally specific LLDP TLVs are also advertised to support LLDP-MED.

- Port VLAN ID TLV (IEEE 802.1 organizationally specific TLVs)
- MAC/PHY configuration/status TLV (IEEE 802.3 organizationally specific TLVs)

LLDP and Cisco Device Stacks

A device stack appears as a single device in the network. Therefore, LLDP discovers the device stack, not the individual stack members.

LLDP and Cisco Medianet

When you configure LLDP or CDP location information on a per-port basis, remote devices can send Cisco Medianet location information to the device.

LLDP-MED

LLDP for Media Endpoint Devices (LLDP-MED) is an extension to LLDP that operates between endpoint devices such as IP phones and network devices. It specifically provides support for voice over IP (VoIP) applications and provides additional TLVs for capabilities discovery, network policy, Power over Ethernet, inventory management and location information. By default, all LLDP-MED TLVs are enabled.

LLDP-MED Supported TLVs

LLDP-MED supports these TLVs:

- LLDP-MED capabilities TLV

Allows LLDP-MED endpoints to determine the capabilities that the connected device supports and has enabled.

- Network policy TLV

Allows both network connectivity devices and endpoints to advertise VLAN configurations and associated Layer 2 and Layer 3 attributes for the specific application on that port. For example, the switch can notify a phone of the VLAN number that it should use. The phone can connect to any device, obtain its VLAN number, and then start communicating with the call control.

By defining a network-policy profile TLV, you can create a profile for voice and voice-signaling by specifying the values for VLAN, class of service (CoS), differentiated services code point (DSCP), and tagging mode. These profile attributes are then maintained centrally on the switch and propagated to the phone.

- Power management TLV

Enables advanced power management between LLDP-MED endpoint and network connectivity devices. Allows devices and phones to convey power information, such as how the device is powered, power priority, and how much power the device needs.

LLDP-MED also supports an extended power TLV to advertise fine-grained power requirements, end-point power priority, and end-point and network connectivity-device power status. LLDP is enabled and power is applied to a port, the power TLV determines the actual power requirement of the endpoint device so that the system power budget can be adjusted accordingly. The device processes the requests and either grants or denies power based on the current power budget. If the request is granted, the switch updates the power budget. If the request is denied, the device turns off power to the port, generates a syslog message, and updates the power budget. If LLDP-MED is disabled or if the endpoint does not support the LLDP-MED power TLV, the initial allocation value is used throughout the duration of the connection.

You can change power settings by entering the **power inline** {**auto** [**max** *max-wattage*] | **never** | **static** [**max** *max-wattage*]} interface configuration command. By default the PoE interface is in **auto** mode; If no value is specified, the maximum is allowed (30 W).

- Inventory management TLV

Allows an endpoint to send detailed inventory information about itself to the device, including information hardware revision, firmware version, software version, serial number, manufacturer name, model name, and asset ID TLV.

- Location TLV

Provides location information from the device to the endpoint device. The location TLV can send this information:

- Civic location information

Provides the civic address information and postal information. Examples of civic location information are street address, road name, and postal community name information.

- ELIN location information

Provides the location information of a caller. The location is determined by the Emergency location identifier number (ELIN), which is a phone number that routes an emergency call to the local public safety answering point (PSAP) and which the PSAP can use to call back the emergency caller.

Wired Location Service

The device uses the location service feature to send location and attachment tracking information for its connected devices to a Cisco Mobility Services Engine (MSE). The tracked device can be a wireless endpoint, a wired endpoint, or a wired device or controller. The device notifies the MSE of device link up and link down events through the Network Mobility Services Protocol (NMSP) location and attachment notifications.

The MSE starts the NMSP connection to the device, which opens a server port. When the MSE connects to the device there are a set of message exchanges to establish version compatibility and service exchange information followed by location information synchronization. After connection, the device periodically sends location and attachment notifications to the MSE. Any link up or link down events detected during an interval are aggregated and sent at the end of the interval.

When the device determines the presence or absence of a device on a link-up or link-down event, it obtains the client-specific information such as the MAC address, IP address, and username. If the client is LLDP-MED- or CDP-capable, the device obtains the serial number and UDI through the LLDP-MED location TLV or CDP.

Depending on the device capabilities, the device obtains this client information at link up:

- Slot and port specified in port connection

- MAC address specified in the client MAC address
- IP address specified in port connection
- 802.1X username if applicable
- Device category is specified as a *wired station*
- State is specified as *new*
- Serial number, UDI
- Model number
- Time in seconds since the device detected the association

Depending on the device capabilities, the device obtains this client information at link down:

- Slot and port that was disconnected
- MAC address
- IP address
- 802.1X username if applicable
- Device category is specified as a *wired station*
- State is specified as *delete*
- Serial number, UDI
- Time in seconds since the device detected the disassociation

When the device shuts down, it sends an attachment notification with the state *delete* and the IP address before closing the NMSP connection to the MSE. The MSE interprets this notification as disassociation for all the wired clients associated with the device.

If you change a location address on the device, the device sends an NMSP location notification message that identifies the affected ports and the changed address information.

Default LLDP Configuration

Table 5: Default LLDP Configuration

Feature	Default Setting
LLDP global state	Disabled
LLDP holdtime (before discarding)	120 seconds
LLDP timer (packet update frequency)	30 seconds
LLDP reinitialization delay	2 seconds
LLDP tlv-select	Disabled to send and receive all TLVs
LLDP interface state	Disabled

Feature	Default Setting
LLDP receive	Disabled
LLDP transmit	Disabled
LLDP med-tlv-select	Disabled to send all LLDP-MED TLVs. When LLDP is globally enabled, LLDP-MED-TLV is also enabled.

Restrictions for LLDP

- If the interface is configured as a tunnel port, LLDP is automatically disabled.
- If you first configure a network-policy profile on an interface, you cannot apply the **switchport voice vlan** command on the interface. If the **switchport voice vlan *vlan-id*** is already configured on an interface, you can apply a network-policy profile on the interface. This way the interface has the voice or voice-signaling VLAN network-policy profile applied on the interface.
- You cannot configure static secure MAC addresses on an interface that has a network-policy profile.
- When Cisco Discovery Protocol and LLDP are both in use within the same switch, it is necessary to disable LLDP on interfaces where Cisco Discovery Protocol is in use for power negotiation. LLDP can be disabled at interface level with the commands **no lldp tlv-select power-management** or **no lldp transmit / no lldp receive**.

How to Configure LLDP, LLDP-MED, and Wired Location Service

Enabling LLDP

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **lldp run**
4. **interface *interface-id***
5. **lldp transmit**
6. **lldp receive**
7. **end**
8. **show lldp**
9. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example:	Enables privileged EXEC mode. • Enter your password if prompted.

	Command or Action	Purpose
	Device> enable	
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	lldp run Example: Device (config)# lldp run	Enables LLDP globally on the device.
Step 4	interface interface-id Example: Device (config)# interface gigabitethernet 2/0/1	Specifies the interface on which you are enabling LLDP, and enter interface configuration mode.
Step 5	lldp transmit Example: Device(config-if)# lldp transmit	Enables the interface to send LLDP packets.
Step 6	lldp receive Example: Device(config-if)# lldp receive	Enables the interface to receive LLDP packets.
Step 7	end Example: Device(config-if)# end	Returns to privileged EXEC mode.
Step 8	show lldp Example: Device# show lldp	Verifies the configuration.
Step 9	copy running-config startup-config Example:	(Optional) Saves your entries in the configuration file.

	Command or Action	Purpose
	Device# <code>copy running-config startup-config</code>	

Configuring LLDP Characteristics

You can configure the frequency of LLDP updates, the amount of time to hold the information before discarding it, and the initialization delay time. You can also select the LLDP and LLDP-MED TLVs to send and receive.



Note Steps 3 through 6 are optional and can be performed in any order.

SUMMARY STEPS

1. `enable`
2. `configure terminal`
3. `lldp holdtime seconds`
4. `lldp reinit delay`
5. `lldp timer rate`
6. `lldp tlv-select`
7. `interface interface-id`
8. `lldp med-tlv-select`
9. `end`
10. `show lldp`
11. `copy running-config startup-config`

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p><code>enable</code></p> <p>Example:</p> <pre>Device> enable</pre>	<p>Enables privileged EXEC mode.</p> <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	<p><code>configure terminal</code></p> <p>Example:</p> <pre>Device# configure terminal</pre>	<p>Enters global configuration mode.</p>
Step 3	<p><code>lldp holdtime seconds</code></p> <p>Example:</p> <pre>Device(config)# lldp holdtime 120</pre>	<p>(Optional) Specifies the amount of time a receiving device should hold the information from your device before discarding it.</p> <p>The range is 0 to 65535 seconds; the default is 120 seconds.</p>

	Command or Action	Purpose
Step 4	lldp reinit <i>delay</i> Example: Device (config)# lldp reinit 2	(Optional) Specifies the delay time in seconds for LLDP to initialize on an interface. The range is 2 to 5 seconds; the default is 2 seconds.
Step 5	lldp timer <i>rate</i> Example: Device (config)# lldp timer 30	(Optional) Sets the sending frequency of LLDP updates in seconds. The range is 5 to 65534 seconds; the default is 30 seconds.
Step 6	lldp tlv-select Example: Device (config)# tlv-select	(Optional) Specifies the LLDP TLVs to send or receive.
Step 7	interface <i>interface-id</i> Example: Device (config)# interface gigabitethernet 2/0/1	Specifies the interface on which you are enabling LLDP, and enter interface configuration mode.
Step 8	lldp med-tlv-select Example: Device (config-if)# lldp med-tlv-select inventory management	(Optional) Specifies the LLDP-MED TLVs to send or receive.
Step 9	end Example: Device (config-if)# end	Returns to privileged EXEC mode.
Step 10	show lldp Example: Device# show lldp	Verifies the configuration.
Step 11	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Configuring LLDP-MED TLVs

By default, the device only sends LLDP packets until it receives LLDP-MED packets from the end device. It then sends LLDP packets with MED TLVs, as well. When the LLDP-MED entry has been aged out, it again only sends LLDP packets.

By using the **lldp** interface configuration command, you can configure the interface not to send the TLVs listed in the following table.

Table 6: LLDP-MED TLVs

LLDP-MED TLV	Description
inventory-management	LLDP-MED inventory management TLV
location	LLDP-MED location TLV
network-policy	LLDP-MED network policy TLV
power-management	LLDP-MED power management TLV

Follow these steps to enable a TLV on an interface:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **lldp med-tlv-select**
5. **end**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	interface <i>interface-id</i> Example: Device (config)# interface	Specifies the interface on which you are enabling LLDP, and enter interface configuration mode.

	Command or Action	Purpose
	<code>gigabitethernet 2/0/1</code>	
Step 4	lldp med-tlv-select Example: <pre>Device(config-if)# lldp med-tlv-select inventory management</pre>	Specifies the TLV to enable.
Step 5	end Example: <pre>Device(config-if)# end</pre>	Returns to privileged EXEC mode.
Step 6	copy running-config startup-config Example: <pre>Device# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Configuring Network-Policy TLV

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **network-policy profile** *profile number*
4. **{voice | voice-signaling} vlan** [*vlan-id* {**cos** *cvalue* | **dscp** *dvalue*}] | [[**dot1p** {**cos** *cvalue* | **dscp** *dvalue*}] | **none** | **untagged**]
5. **exit**
6. **interface** *interface-id*
7. **network-policy** *profile number*
8. **lldp med-tlv-select network-policy**
9. **end**
10. **show network-policy profile**
11. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example:	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.

	Command or Action	Purpose
	Device> enable	
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	network-policy profile <i>profile number</i> Example: Device(config)# network-policy profile 1	Specifies the network-policy profile number, and enter network-policy configuration mode. The range is 1 to 4294967295.
Step 4	{voice voice-signaling} vlan [<i>vlan-id</i> { <i>cos cvalue</i> dscp dvalue }] [[dot1p { <i>cos cvalue</i> dscp dvalue }] none untagged] Example: Device(config-network-policy)# voice vlan 100 cos 4	Configures the policy attributes: <ul style="list-style-type: none"> • voice—Specifies the voice application type. • voice-signaling—Specifies the voice-signaling application type. • vlan—Specifies the native VLAN for voice traffic. • <i>vlan-id</i>—(Optional) Specifies the VLAN for voice traffic. The range is 1 to 4094. • <i>cos cvalue</i>—(Optional) Specifies the Layer 2 priority class of service (CoS) for the configured VLAN. The range is 0 to 7; the default is 5. • dscp dvalue—(Optional) Specifies the differentiated services code point (DSCP) value for the configured VLAN. The range is 0 to 63; the default is 46. • dot1p—(Optional) Configures the telephone to use IEEE 802.1p priority tagging and use VLAN 0 (the native VLAN). • none—(Optional) Do not instruct the IP telephone about the voice VLAN. The telephone uses the configuration from the telephone key pad. • untagged—(Optional) Configures the telephone to send untagged voice traffic. This is the default for the telephone. • untagged—(Optional) Configures the telephone to send untagged voice traffic. This is the default for the telephone.

	Command or Action	Purpose
Step 5	exit Example: Device (config) # exit	Returns to global configuration mode.
Step 6	interface <i>interface-id</i> Example: Device (config) # interface gigabitethernet 2/0/1	Specifies the interface on which you are configuring a network-policy profile, and enter interface configuration mode.
Step 7	network-policy <i>profile number</i> Example: Device (config-if) # network-policy 1	Specifies the network-policy profile number.
Step 8	lldp med-tlv-select network-policy Example: Device (config-if) # lldp med-tlv-select network-policy	Specifies the network-policy TLV.
Step 9	end Example: Device (config) # end	Returns to privileged EXEC mode.
Step 10	show network-policy profile Example: Device# show network-policy profile	Verifies the configuration.
Step 11	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Configuring Location TLV and Wired Location Service

Beginning in privileged EXEC mode, follow these steps to configure location information for an endpoint and to apply it to an interface.

SUMMARY STEPS

1. **configure terminal**
2. **location** {**admin-tag** *string* | **civic-location identifier** {*id* | **host**} | **elin-location** *string identifier id* | **custom-location identifier** {*id* | **host**} | **geo-location identifier** {*id* | **host**}}
3. **exit**
4. **interface** *interface-id*
5. **location** {**additional-location-information** *word* | **civic-location-id** {*id* | **host**} | **elin-location-id** *id* | **custom-location-id** {*id* | **host**} | **geo-location-id** {*id* | **host**}}
6. **end**
7. Use one of the following:
 - **show location admin-tag** *string*
 - **show location civic-location identifier** *id*
 - **show location elin-location identifier** *id*
8. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 2	location { admin-tag <i>string</i> civic-location identifier { <i>id</i> host } elin-location <i>string identifier id</i> custom-location identifier { <i>id</i> host } geo-location identifier { <i>id</i> host }} Example: Device(config)# location civic-location identifier 1 Device(config-civic)# number 3550 Device(config-civic)# primary-road-name "Cisco Way" Device(config-civic)# city "San Jose" Device(config-civic)# state CA Device(config-civic)# building 19 Device(config-civic)# room C6 Device(config-civic)# county "Santa Clara" Device(config-civic)# country US	Specifies the location information for an endpoint. <ul style="list-style-type: none"> • admin-tag—Specifies an administrative tag or site information. • civic-location—Specifies civic location information. • elin-location—Specifies emergency location information (ELIN). • custom-location—Specifies custom location information. • geo-location—Specifies geo-spatial location information. • identifier <i>id</i>—Specifies the ID for the civic, ELIN, custom, or geo location. • host—Specifies the host civic, custom, or geo location. • <i>string</i>—Specifies the site or location information in alphanumeric format.
Step 3	exit Example:	Returns to global configuration mode.

	Command or Action	Purpose
	Device(config-civic)# exit	
Step 4	interface <i>interface-id</i> Example: Device (config)# interface gigabitethernet2/0/1	Specifies the interface on which you are configuring the location information, and enter interface configuration mode.
Step 5	location { additional-location-information <i>word</i> civic-location-id { <i>id</i> host } elin-location-id <i>id</i> custom-location-id { <i>id</i> host } geo-location-id { <i>id</i> host } } Example: Device(config-if)# location elin-location-id 1	Enters location information for an interface: <ul style="list-style-type: none"> • additional-location-information—Specifies additional information for a location or place. • civic-location-id—Specifies global civic location information for an interface. • elin-location-id—Specifies emergency location information for an interface. • custom-location-id—Specifies custom location information for an interface. • geo-location-id—Specifies geo-spatial location information for an interface. • host—Specifies the host location identifier. • <i>word</i>—Specifies a word or phrase with additional location information. • <i>id</i>—Specifies the ID for the civic, ELIN, custom, or geo location. The ID range is 1 to 4095.
Step 6	end Example: Device(config-if)# end	Returns to privileged EXEC mode.
Step 7	Use one of the following: <ul style="list-style-type: none"> • show location admin-tag <i>string</i> • show location civic-location identifier <i>id</i> • show location elin-location identifier <i>id</i> Example: Device# show location admin-tag or Device# show location civic-location	Verifies the configuration.

	Command or Action	Purpose
	<pre>identifier</pre> <p>OR</p> <pre>Device# show location elin-location identifier</pre>	
Step 8	<p>copy running-config startup-config</p> <p>Example:</p> <pre>Device# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Enabling Wired Location Service on the Device

Before you begin

For wired location to function, you must first enter the **ip device tracking** global configuration command.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **nmosp notification interval {attachment | location} interval-seconds**
4. **end**
5. **show network-policy profile**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p>enable</p> <p>Example:</p> <pre>Device> enable</pre>	<p>Enables privileged EXEC mode.</p> <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	<p>configure terminal</p> <p>Example:</p> <pre>Device# configure terminal</pre>	Enters global configuration mode.
Step 3	<p>nmosp notification interval {attachment location} interval-seconds</p> <p>Example:</p>	<p>Specifies the NMSP notification interval.</p> <p>attachment—Specifies the attachment notification interval.</p>

	Command or Action	Purpose
	Device(config)# nmsp notification interval location 10	location —Specifies the location notification interval. <i>interval-seconds</i> —Duration in seconds before the device sends the MSE the location or attachment updates. The range is 1 to 30; the default is 30.
Step 4	end Example: Device(config)# end	Returns to privileged EXEC mode.
Step 5	show network-policy profile Example: Device# show network-policy profile	Verifies the configuration.
Step 6	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Configuration Examples for LLDP, LLDP-MED, and Wired Location Service

Configuring Network-Policy TLV: Examples

This example shows how to configure VLAN 100 for voice application with CoS and to enable the network-policy profile and network-policy TLV on an interface:

```
# configure terminal
(config)# network-policy 1
(config-network-policy)# voice vlan 100 cos 4
(config-network-policy)# exit
(config)# interface gigabitethernet 1/0/1
(config-if)# network-policy profile 1
(config-if)# lldp med-tlv-select network-policy
```

This example shows how to configure the voice application type for the native VLAN with priority tagging:

```
config-network-policy)# voice vlan dot1p cos 4
config-network-policy)# voice vlan dot1p dscp 34
```

Monitoring and Maintaining LLDP, LLDP-MED, and Wired Location Service

Commands for monitoring and maintaining LLDP, LLDP-MED, and wired location service.

Command	Description
clear lldp counters	Resets the traffic counters to zero.
clear lldp table	Deletes the LLDP neighbor information table.
clear nmsp statistics	Clears the Nmsp statistic counters.
show lldp	Displays global information, such as frequency of transmissions, the holdtime for packets being sent, and the delay time before LLDP initializes on an interface.
show lldp entry <i>entry-name</i>	Displays information about a specific neighbor. You can enter an asterisk (*) to display all neighbors, or you can enter the neighbor name.
show lldp interface [<i>interface-id</i>]	Displays information about interfaces with LLDP enabled. You can limit the display to a specific interface.
show lldp neighbors [<i>interface-id</i>] [detail]	Displays information about neighbors, including device type, interface type and number, holdtime settings, capabilities, and port ID. You can limit the display to neighbors of a specific interface or expand the display for more detailed information.
show lldp traffic	Displays LLDP counters, including the number of packets sent and received, number of packets discarded, and number of unrecognized TLVs.
show location admin-tag <i>string</i>	Displays the location information for the specified administrative tag or site.
show location civic-location identifier <i>id</i>	Displays the location information for a specific global civic location.
show location elin-location identifier <i>id</i>	Displays the location information for an emergency location
show network-policy profile	Displays the configured network-policy profiles.
show nmsp	Displays the Nmsp information

Additional References for LLDP, LLDP-MED, and Wired Location Service

MIBs

MIB	MIBs Link
All the supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
<p>The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies.</p> <p>To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds.</p> <p>Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.</p>	http://www.cisco.com/support

Feature Information for LLDP, LLDP-MED, and Wired Location Service

Release	Modification
Cisco IOS Release 15.0(2)EX1	This feature was introduced.



CHAPTER 5

Configuring System MTU

- [Information About the MTU, on page 59](#)
- [How to Configure MTU , on page 59](#)
- [Configuration Examples for System MTU, on page 60](#)
- [Additional References for System MTU, on page 61](#)
- [Feature Information for System MTU, on page 61](#)

Information About the MTU

The default maximum transmission unit (MTU) size for frames received and sent on all device interfaces is 1500 bytes.

You can change the MTU size to support switched jumbo frames on all Gigabit Ethernet and 10-Gigabit Ethernet interfaces and to support routed frames on all routed ports.

Restrictions for System MTU

When configuring the system MTU values, follow these guidelines:

- The device does not support the MTU on a per-interface basis.

How to Configure MTU

Configuring the System MTU

Beginning in privileged EXEC mode, follow these steps to change the MTU size for switched and routed packets:

SUMMARY STEPS

1. **configure terminal**
2. **system mtu *bytes***
3. **system mtu jumbo *bytes***
4. **system mtu routing *bytes***

5. `end`
6. `copy running-config startup-config`
7. `show system mtu`

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode.
Step 2	system mtu bytes Example: Device(config)# <code>system mtu 2500</code>	(Optional) Changes the MTU size for all Fast Ethernet interfaces on the switch. The range is 1500 to 9198 bytes; the default is 1500 bytes.
Step 3	system mtu jumbo bytes Example: Device(config)# <code>system mtu jumbo 7500</code>	(Optional) Changes the MTU size for all Gigabit Ethernet and 10-Gigabit Ethernet interfaces on the switch or the switch stack. The range is 1500 to 9198 bytes; the default is 1500 bytes.
Step 4	system mtu routing bytes Example: Device(config)# <code>system mtu routing 2000</code>	(Optional) Changes the system MTU for routed ports. You can also set the maximum MTU to be advertised by the routing protocols that support the configured MTU size. The system routing MTU is the maximum MTU for routed packets and is also the maximum MTU that the switch advertises in routing updates for protocols such as OSPF. Note This command is not supported on switches running the LAN base feature set.
Step 5	end Example: Device(config)# <code>end</code>	Returns to privileged EXEC mode.
Step 6	copy running-config startup-config Example: Device# <code>copy running-config startup-config</code>	Saves your entries in the configuration file.
Step 7	show system mtu Example: Device# <code>show system mtu</code>	Verifies your settings.

Configuration Examples for System MTU

This example shows how to set the maximum packet size for a Gigabit Ethernet port to 7500 bytes:

```
Device(config)# system mtu 7500
Device(config)# exit
```

This example shows how to set the jumbo packet size for a Gigabit Ethernet port to 7500 bytes:

```
Device(config)# system mtu jumbo 7500
Device(config)# exit
```

If you enter a value that is outside the allowed range for the specific type of interface, the value is not accepted. This example shows the response when you try to set Gigabit Ethernet interfaces to an out-of-range number:

```
Device(config)# system mtu jumbo 25000
                        ^
% Invalid input detected at '^' marker.
```

Additional References for System MTU

MIBs

MIB	MIBs Link
All the supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
<p>The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies.</p> <p>To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds.</p> <p>Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.</p>	http://www.cisco.com/support

Feature Information for System MTU

Release	Modification
Cisco IOS Release 15.0(2)EX1	This feature was introduced.



CHAPTER 6

Configuring Internal Power Supplies

- [Information About Internal Power Supplies](#) , on page 63
- [How to Configure Internal Power Supplies](#), on page 63
- [Monitoring Internal Power Supplies](#), on page 64
- [Configuration Examples for Internal Power Supplies](#), on page 64
- [Additional References](#), on page 65
- [Feature History and Information for Internal Power Supplies](#), on page 65

Information About Internal Power Supplies

See the device installation guide for information about the power supplies.

How to Configure Internal Power Supplies

Configuring Internal Power Supply

You can use the **power supply** EXEC command to configure and manage the internal power supply on the device. The device does not support the **no power supply** EXEC command.

Follow these steps beginning in user EXEC mode:

SUMMARY STEPS

1. **power supply** *switch_number* **slot**{**A** | **B**} { **off** | **on** }
2. **show environment power**

DETAILED STEPS

	Command or Action	Purpose
Step 1	power supply <i>switch_number</i> slot { A B } { off on } Example: Device# power supply 1 slot A on	Sets the specified power supply to off or on by using one of these keywords: <ul style="list-style-type: none">• A —Selects the power supply in slot A.• B —Selects power supply in slot B.

	Command or Action	Purpose
		<p>Note Power supply slot B is the closest to the outer edge of the device.</p> <ul style="list-style-type: none"> • off —Set the power supply off. • on —Set the power supply on. <p>By default, the device power supply is on.</p>
Step 2	<p>show environment power</p> <p>Example:</p> <pre>Device# show environment power</pre>	Verifies your settings.

Monitoring Internal Power Supplies

Table 7: Show Commands for Power Supplies

Command	Purpose
<p>show environment power [all switch <i>switch_number</i>]</p>	<p>(Optional) Displays the status of the internal power supplies for the specified device. .</p> <p>The device keywords are available only on stacking-capable devices.</p>

Configuration Examples for Internal Power Supplies

This example shows how to set the power supply in slot A to off:

```
Device# power supply 1 slot A off
Disabling Power supply A may result in a power loss to PoE devices and/or switches ...
Continue? (yes/[no]): yes
Device#
Jun 10 04:52:54.389: %PLATFORM_ENV-6-FRU_PS_OIR: FRU Power Supply 1 powered off
Jun 10 04:52:56.717: %PLATFORM_ENV-1-FAN_NOT_PRESENT: Fan is not present
Device#
```

This example shows how to set the power supply in slot A to on:

```
Device# power supply 1 slot A on
Jun 10 04:54:39.600: %PLATFORM_ENV-6-FRU_PS_OIR: FRU Power Supply 1 powered on
```

This example shows the output of the **show env power** command:

Table 8: show env power Status Descriptions

Field	Description
OK	The power supply is present and power is good.
Not Present	No power supply is installed.
No Input Power	The power supply is present but there is no input power.
Disabled	The power supply and input power are present, but power supply is switched off by CLI.
Not Responding	The power supply is not recognizable or is faulty.
Failure-Fan	The power supply fan is faulty.

Additional References

MIBs

MIB	MIBs Link
All the supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
<p>The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies.</p> <p>To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds.</p> <p>Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.</p>	http://www.cisco.com/support

Feature History and Information for Internal Power Supplies

Release	Modification
Cisco IOS Release 15.0(2)EX1	This feature was introduced.



CHAPTER 7

Configuring Boot Fast

- [Configuring Boot Fast on the switch, on page 67](#)

Configuring Boot Fast on the switch

This feature, when enabled, helps the switch to boot up fast. The Memory test is performed for a limited range, the switch skips File system check (FSCK) and Skips Post test.



Note When Fast boot is enabled, you can still run the POST tests manually from the command line interface, once the switch has booted up, using **diagnostic start** command.

Enabling Boot Fast

To enable the boot fast feature, perform the following steps:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **boot fast**
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none">• Enter your password if prompted.
Step 2	configure terminal Example:	Enters global configuration mode.

	Command or Action	Purpose
	Device# <code>configure terminal</code>	
Step 3	boot fast Example: Device(config)# <code>boot fast</code>	Enables fast boot feature Performs Memory test for a limited range, Skips File system check (FSCK) and Skips Post test.
Step 4	end Example: Device(config)# <code>end</code>	Returns to privileged EXEC mode.

Disabling Boot Fast

To disable the boot fast feature, perform the following steps:

SUMMARY STEPS

1. `enable`
2. `configure terminal`
3. `no boot fast`
4. `end`

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> <code>enable</code>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode.
Step 3	no boot fast Example: Device(config)# <code>no boot fast</code>	Disables the boot fast feature.

	Command or Action	Purpose
Step 4	end Example: Device (config) # end	Returns to privileged EXEC mode.



CHAPTER 8

Configuring PoE

- [Information About PoE, on page 71](#)
- [How to Configure PoE, on page 76](#)
- [Monitoring Power Status, on page 83](#)
- [Configuration Examples for Configuring PoE, on page 83](#)
- [Additional References, on page 83](#)

Information About PoE

Power over Ethernet Ports

A PoE-capable switch port automatically supplies power to one of these connected devices if the device senses that there is no power on the circuit:

- a Cisco pre-standard powered device (such as a Cisco IP Phone or a Cisco Aironet Access Point)
- an IEEE 802.3af-compliant powered device

A powered device can receive redundant power when it is connected to a PoE switch port and to an AC power source. The device does not receive redundant power when it is only connected to the PoE port.

Supported Protocols and Standards

The device uses these protocols and standards to support PoE:

- CDP with power consumption—The powered device notifies the device of the amount of power it is consuming. The device does not reply to the power-consumption messages. The device can only supply power to or remove power from the PoE port.
- Cisco intelligent power management—The powered device and the device negotiate through power-negotiation CDP messages for an agreed-upon power-consumption level. The negotiation allows a high-power Cisco powered device, which consumes more than 7 W, to operate at its highest power mode. The powered device first boots up in low-power mode, consumes less than 7 W, and negotiates to obtain enough power to operate in high-power mode. The device changes to high-power mode only when it receives confirmation from the device.

High-power devices can operate in low-power mode on devices that do not support power-negotiation CDP.

Cisco intelligent power management is backward-compatible with CDP with power consumption; the device responds according to the CDP message that it receives. CDP is not supported on third-party powered devices; therefore, the device uses the IEEE classification to determine the power usage of the device.

- IEEE 802.3af—The major features of this standard are powered-device discovery, power administration, disconnect detection, and optional powered-device power classification. For more information, see the standard.

Powered-Device Detection and Initial Power Allocation

The device detects a Cisco pre-standard or an IEEE-compliant powered device when the PoE-capable port is in the no-shutdown state, PoE is enabled (the default), and the connected device is not being powered by an AC adaptor.

After device detection, the device determines the device power requirements based on its type:

- The initial power allocation is the maximum amount of power that a powered device requires. The device initially allocates this amount of power when it detects and powers the powered device. As the device receives CDP messages from the powered device and as the powered device negotiates power levels with the device through CDP power-negotiation messages, the initial power allocation might be adjusted.
- The device classifies the detected IEEE device within a power consumption class. Based on the available power in the power budget, the device determines if a port can be powered. [Table 9: IEEE Power Classifications, on page 72](#) lists these levels.

Table 9: IEEE Power Classifications

Class	Maximum Power Level Required from the Device
0 (class status unknown)	15.4 W
1	4 W
2	7 W
3	15.4 W
4	30 W (For IEEE 802.3at Type 2 powered devices)

The device monitors and tracks requests for power and grants power only when it is available. The device tracks its power budget (the amount of power available on the device for PoE). The device performs power-accounting calculations when a port is granted or denied power to keep the power budget up to date.

After power is applied to the port, the device uses CDP to determine the *CDP-specific* power consumption requirement of the connected Cisco powered devices, which is the amount of power to allocate based on the CDP messages. The device adjusts the power budget accordingly. This does not apply to third-party PoE devices. The device processes a request and either grants or denies power. If the request is granted, the device updates the power budget. If the request is denied, the device ensures that power to the port is turned off, generates a syslog message, and updates the LEDs. Powered devices can also negotiate with the device for more power.

With PoE+, powered devices use IEEE 802.3at and LLDP power with media dependent interface (MDI) type, length, and value descriptions (TLVs), Power-via-MDI TLVs, for negotiating power up to 30 W. Cisco

pre-standard devices and Cisco IEEE powered devices can use CDP or the IEEE 802.3at power-via-MDI power negotiation mechanism to request power levels up to 30 W.



Note The initial allocation for Class 0, Class 3, and Class 4 powered devices is 15.4 W. When a device starts up and uses CDP or LLDP to send a request for more than 15.4 W, it can be allocated up to the maximum of 30 W.



Note The CDP-specific power consumption requirement is referred to as the *actual* power consumption requirement in the software configuration guides and command references.

If the device detects a fault caused by an undervoltage, overvoltage, overtemperature, oscillator-fault, or short-circuit condition, it turns off power to the port, generates a syslog message, and updates the power budget and LEDs.

The PoE feature operates the same whether or not the device is a stack member. The power budget is per device and independent of any other device in the stack. Election of a new active device does not affect PoE operation. The active device keeps track of the PoE status for all devices and ports in the stack and includes the status in output displays.

Power Management Modes

The device supports these PoE modes:

- **auto**—The device automatically detects if the connected device requires power. If the device discovers a powered device connected to the port and if the device has enough power, it grants power, updates the power budget, turns on power to the port on a first-come, first-served basis, and updates the LEDs. For LED information, see the hardware installation guide.

If the device has enough power for all the powered devices, they all come up. If enough power is available for all powered devices connected to the device, power is turned on to all devices. If there is not enough available PoE, or if a device is disconnected and reconnected while other devices are waiting for power, it cannot be determined which devices are granted or are denied power.

If granting power would exceed the system power budget, the device denies power, ensures that power to the port is turned off, generates a syslog message, and updates the LEDs. After power has been denied, the device periodically rechecks the power budget and continues to attempt to grant the request for power.

If a device being powered by the device is then connected to wall power, the device might continue to power the device. The device might continue to report that it is still powering the device whether the device is being powered by the device or receiving power from an AC power source.

If a powered device is removed, the device automatically detects the disconnect and removes power from the port. You can connect a nonpowered device without damaging it.

You can specify the maximum wattage that is allowed on the port. If the IEEE class maximum wattage of the powered device is greater than the configured maximum value, the device does not provide power to the port. If the device powers a powered device, but the powered device later requests through CDP messages more than the configured maximum value, the device removes power to the port. The power that was allocated to the powered device is reclaimed into the global power budget. If you do not specify a wattage, the device delivers the maximum value. Use the **auto** setting on any PoE port. The auto mode is the default setting.

- **static**—The device pre-allocates power to the port (even when no powered device is connected) and guarantees that power will be available for the port. The device allocates the port configured maximum wattage, and the amount is never adjusted through the IEEE class or by CDP messages from the powered device. Because power is pre-allocated, any powered device that uses less than or equal to the maximum wattage is guaranteed to be powered when it is connected to the static port. The port no longer participates in the first-come, first-served model.

However, if the powered-device IEEE class is greater than the maximum wattage, the device does not supply power to it. If the device learns through CDP messages that the powered device is consuming more than the maximum wattage, the device shuts down the powered device.

If you do not specify a wattage, the device pre-allocates the maximum value. The device powers the port only if it discovers a powered device. Use the **static** setting on a high-priority interface.

- **never**—The device disables powered-device detection and never powers the PoE port even if an unpowered device is connected. Use this mode only when you want to make sure that power is never applied to a PoE-capable port, making the port a data-only port.

For most situations, the default configuration (auto mode) works well, providing plug-and-play operation. No further configuration is required. However, perform this task to configure a PoE port for a higher priority, to make it data only, or to specify a maximum wattage to disallow high-power powered devices on a port.

Power Monitoring and Power Policing

When policing of the real-time power consumption is enabled, the device takes action when a powered device consumes more power than the maximum amount allocated, also referred to as the *cutoff-power value*.

When PoE is enabled, the device senses the real-time power consumption of the powered device. The device monitors the real-time power consumption of the connected powered device; this is called *power monitoring* or *power sensing*. The device also polices the power usage with the *power policing* feature.

Power monitoring is backward-compatible with Cisco intelligent power management and CDP-based power consumption. It works with these features to ensure that the PoE port can supply power to the powered device.

The device senses the real-time power consumption of the connected device as follows:

1. The device monitors the real-time power consumption on individual ports.
2. The device records the power consumption, including peak power usage. The device reports the information through the CISCO-POWER-ETHERNET-EXT-MIB.
3. If power policing is enabled, the device polices power usage by comparing the real-time power consumption to the maximum power allocated to the device. The maximum power consumption is also referred to as the *cutoff power* on a PoE port.

If the device uses more than the maximum power allocation on the port, the device can either turn off power to the port, or the device can generate a syslog message and update the LEDs (the port LED is now blinking amber) while still providing power to the device based on the device configuration. By default, power-usage policing is disabled on all PoE ports.

If error recovery from the PoE error-disabled state is enabled, the device automatically takes the PoE port out of the error-disabled state after the specified amount of time.

If error recovery is disabled, you can manually re-enable the PoE port by using the **shutdown** and **no shutdown** interface configuration commands.

4. If policing is disabled, no action occurs when the powered device consumes more than the maximum power allocation on the PoE port, which could adversely affect the device.

Maximum Power Allocation (Cutoff Power) on a PoE Port

When power policing is enabled, the device determines one of these values as the cutoff power on the PoE port in this order:

1. Manually when you set the user-defined power level that the device budgets for the port by using the **power inline consumption default** *wattage* global or interface configuration command
2. Manually when you set the user-defined power level that limits the power allowed on the port by using the **power inline auto max** *max-wattage* or the **power inline static max** *max-wattage* interface configuration command
3. Automatically when the device sets the power usage of the device by using CDP power negotiation .

Use the first or second method in the previous list to manually configure the cutoff-power value by entering the **power inline consumption default** *wattage* or the **power inline [auto | static max]** *max-wattage* command.

You should use **power inline consumption default** *wattage* command to manually set the power level for a port only in situations where CDP/LLDP power negotiations are not supported.

Power Consumption Values

You can configure the initial power allocation and the maximum power allocation on a port. However, these values are only the configured values that determine when the device should turn on or turn off power on the PoE port. The maximum power allocation is not the same as the actual power consumption of the powered device. The actual cutoff power value that the device uses for power policing is not equal to the configured power value.

When power policing is enabled, the device polices the power usage *at the switch port*, which is greater than the power consumption of the device. When you manually set the maximum power allocation, you must consider the power loss over the cable from the switch port to the powered device. The cutoff power is the sum of the rated power consumption of the powered device and the worst-case power loss over the cable.

We recommend that you enable power policing when PoE is enabled on your device. For example, if policing is disabled and you set the cutoff-power value by using the **power inline auto max 6300** interface configuration command, the configured maximum power allocation on the PoE port is 6.3 W (6300 mW). The device provides power to the connected devices on the port if the device needs up to 6.3 W. If the CDP-power negotiated value or the IEEE classification value exceeds the configured cutoff value, the device does not provide power to the connected device. After the device turns on power on the PoE port, the device does not police the real-time power consumption of the device, and the device can consume more power than the maximum allocated amount, which could adversely affect the device and the devices connected to the other PoE ports.

How to Configure PoE

Configuring a Power Management Mode on a PoE Port



Note When you make PoE configuration changes, the port being configured drops power. Depending on the new configuration, the state of the other PoE ports, and the state of the power budget, the port might not be powered up again. For example, port 1 is in the auto and on state, and you configure it for static mode. The device removes power from port 1, detects the powered device, and repowers the port. If port 1 is in the auto and on state and you configure it with a maximum wattage of 10 W, the device removes power from the port and then redetects the powered device. The device repowers the port only if the powered device is a class 1, class 2, or a Cisco-only powered device.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **power inline** {**auto** [**max** *max-wattage*] | **never** | **static** [**max** *max-wattage*]}
5. **end**
6. **show power inline** [*interface-id* | **module** *switch-number*]
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	interface <i>interface-id</i> Example: Device(config)# interface gigabitethernet 2/0/1	Specifies the physical port to be configured, and enters interface configuration mode.
Step 4	power inline { auto [max <i>max-wattage</i>] never static [max <i>max-wattage</i>]} Example: Device(config-if)# power inline auto	Configures the PoE mode on the port. The keywords have these meanings: <ul style="list-style-type: none"> • auto—Enables powered-device detection. If enough power is available, automatically allocates power to

	Command or Action	Purpose
		<p>the PoE port after device detection. This is the default setting.</p> <ul style="list-style-type: none"> • max <i>max-wattage</i>—Limits the power allowed on the port. If no value is specified, the maximum is allowed. • never—Disables device detection, and disable power to the port. <p>Note If a port has a Cisco powered device connected to it, do not use the power inline never command to configure the port. A false link-up can occur, placing the port into the error-disabled state.</p> <ul style="list-style-type: none"> • static—Enables powered-device detection. Pre-allocate (reserve) power for a port before the device discovers the powered device. The device reserves power for this port even when no device is connected and guarantees that power will be provided upon device detection. <p>The device allocates power to a port configured in static mode before it allocates power to a port configured in auto mode.</p>
Step 5	<p>end</p> <p>Example:</p> <pre>Device(config-if)# end</pre>	Returns to privileged EXEC mode.
Step 6	<p>show power inline [<i>interface-id</i> module <i>switch-number</i>]</p> <p>Example:</p> <pre>Device# show power inline</pre>	<p>Displays PoE status for a device or a device stack, for the specified interface, or for a specified stack member.</p> <p>The module <i>switch-number</i> keywords are supported only on stacking-capable devices.</p>
Step 7	<p>copy running-config startup-config</p> <p>Example:</p> <pre>Device# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Budgeting Power for Devices Connected to a PoE Port

When Cisco powered devices are connected to PoE ports, the device uses Cisco Discovery Protocol (CDP) to determine the *protocol-specific* power consumption of the devices, and the device adjusts the power budget accordingly. This does not apply to IEEE third-party powered devices. For these devices, when the device grants a power request, the device adjusts the power budget according to the powered-device IEEE classification. If the powered device is a class 0 (class status unknown) or a class 3, the device budgets 15,400 mW for the device, regardless of the CDP-specific amount of power needed. If the powered device reports a higher class

than its CDP-specific consumption or does not support power classification (defaults to class 0), the device can power fewer devices because it uses the IEEE class information to track the global power budget.

By using the **power inline consumption wattage** interface configuration command or the **power inline consumption default wattage** global configuration command, you can override the default power requirement specified by the IEEE classification. The difference between what is mandated by the IEEE classification and what is actually needed by the device is reclaimed into the global power budget for use by additional devices. You can then extend the device power budget and use it more effectively.



Caution You should carefully plan your device power budget, enable the power monitoring feature, and make certain not to oversubscribe the power supply.



Note When you manually configure the power budget, you must also consider the power loss over the cable between the device and the powered device.

Budgeting Power to All PoE ports

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **no cdp run**
4. **power inline consumption default wattage**
5. **end**
6. **show power inline consumption default**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	no cdp run Example: Device(config)# no cdp run	(Optional) Disables CDP.

	Command or Action	Purpose
Step 4	<p>power inline consumption default <i>wattage</i></p> <p>Example:</p> <pre>Device(config)# power inline consumption default 5000</pre>	<p>Configures the power consumption of powered devices connected to each PoE port.</p> <p>The range for each device is 4000 to 30000 mW (PoE+). The default is 30000 mW.</p> <p>Note When you use this command, we recommend you also enable power policing.</p>
Step 5	<p>end</p> <p>Example:</p> <pre>Device(config)# end</pre>	Returns to privileged EXEC mode.
Step 6	<p>show power inline consumption default</p> <p>Example:</p> <pre>Device# show power inline consumption default</pre>	Displays the power consumption status.
Step 7	<p>copy running-config startup-config</p> <p>Example:</p> <pre>Device# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Budgeting Power to a Specific PoE Port

SUMMARY STEPS

1. enable
2. configure terminal
3. no cdp run
4. interface *interface-id*
5. power inline consumption *wattage*
6. end
7. show power inline consumption
8. copy running-config startup-config

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p>enable</p> <p>Example:</p> <pre>Device> enable</pre>	<p>Enables privileged EXEC mode.</p> <ul style="list-style-type: none"> • Enter your password if prompted.

	Command or Action	Purpose
Step 2	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode.
Step 3	no cdp run Example: Device(config)# <code>no cdp run</code>	(Optional) Disables CDP.
Step 4	interface <i>interface-id</i> Example: Device(config)# <code>interface gigabitethernet 1/0/1</code>	Specifies the physical port to be configured, and enter interface configuration mode.
Step 5	power inline consumption <i>wattage</i> Example: Device(config-if)# <code>power inline consumption 5000</code>	Configures the power consumption of a powered device connected to a PoE port on the device. The range for each device is 4000 to 30000 mW (PoE+). The default is 30000 mW (PoE+).
Step 6	end Example: Device(config-if)# <code>end</code>	Returns to privileged EXEC mode.
Step 7	show power inline consumption Example: Device# <code>show power inline consumption</code>	Displays the power consumption data.
Step 8	copy running-config startup-config Example: Device# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Configuring Power Policing

By default, the device monitors the real-time power consumption of connected powered devices. You can configure the device to police the power usage. By default, policing is disabled.

SUMMARY STEPS

1. `enable`
2. `configure terminal`
3. `interface interface-id`
4. `power inline police [action {log | errdisable}]`
5. `exit`

6. Use one of the following:
 - **errdisable detect cause inline-power**
 - **errdisable recovery cause inline-power**
 - **errdisable recovery interval *interval***
7. **exit**
8. Use one of the following:
 - **show power inline police**
 - **show errdisable recovery**
9. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	interface <i>interface-id</i> Example: Device(config)# interface gigabitethernet 2/0/1	Specifies the physical port to be configured, and enter interface configuration mode.
Step 4	power inline police [action{log errdisable}] Example: Device(config-if)# power inline police	If the real-time power consumption exceeds the maximum power allocation on the port, configures the device to take one of these actions: <ul style="list-style-type: none"> • power inline police—Shuts down the PoE port, turns off power to it, and puts it in the error-disabled state. <p>Note You can enable error detection for the PoE error-disabled cause by using the errdisable detect cause inline-power global configuration command. You can also enable the timer to recover from the PoE error-disabled state by using the errdisable recovery cause inline-power interval <i>interval</i> global configuration command.</p> <ul style="list-style-type: none"> • power inline police action errdisable—Turns off power to the port if the real-time power consumption exceeds the maximum power allocation on the port.

	Command or Action	Purpose
		<ul style="list-style-type: none"> • power inline police action log—Generates a syslog message while still providing power to the port. <p>If you do not enter the action log keywords, the default action shuts down the port and puts the port in the error-disabled state.</p>
Step 5	exit Example: Device(config-if) # exit	Returns to global configuration mode.
Step 6	Use one of the following: <ul style="list-style-type: none"> • errdisable detect cause inline-power • errdisable recovery cause inline-power • errdisable recovery interval interval Example: Device(config) # errdisable detect cause inline-power Device(config) # errdisable recovery cause inline-power Device(config) # errdisable recovery interval 100	(Optional) Enables error recovery from the PoE error-disabled state, and configures the PoE recover mechanism variables. By default, the recovery interval is 300 seconds. For interval interval , specifies the time in seconds to recover from the error-disabled state. The range is 30 to 86400.
Step 7	exit Example: Device(config) # exit	Returns to privileged EXEC mode.
Step 8	Use one of the following: <ul style="list-style-type: none"> • show power inline police • show errdisable recovery Example: Device# show power inline police Device# show errdisable recovery	Displays the power monitoring status, and verify the error recovery settings.
Step 9	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Monitoring Power Status

Table 10: Show Commands for Power Status

Command	Purpose
show env power switch [<i>switch-number</i>]	(Optional) Displays the status of the internal power supplies for each switch in the stack or for the specified switch. The range is 1 to 8, depending on the switch member numbers in the stack. These keywords are available only on stacking-capable switches.
show power inline [<i>interface-id</i> module <i>switch-number</i>]	Displays PoE status for a switch or switch stack, for an interface, or for a specific switch in the stack.
show power inline police	Displays the power policing data.

Configuration Examples for Configuring PoE

Budgeting Power: Example

When you enter one of the following commands,

- **[no] power inline consumption default** *wattage* global configuration command
- **[no] power inline consumption** *wattage*
interface configuration command

this caution message appears:

```
%CAUTION: Interface Gi1/0/1: Misconfiguring the 'power inline consumption/allocation'
command may cause damage to the
switch and void your warranty. Take precaution not to oversubscribe the power supply. It
is recommended to enable power
policing if the switch supports it. Refer to documentation.
```

Additional References

MIBs

MIB	MIBs Link
All the supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
<p>The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies.</p> <p>To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds.</p> <p>Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.</p>	http://www.cisco.com/support



CHAPTER 9

Configuring 2-event Classification

- [Information about 2-event Classification, on page 85](#)
- [Configuring 2-event Classification, on page 85](#)
- [Example: Configuring 2-Event Classification, on page 86](#)

Information about 2-event Classification

When a class 4 device gets detected, IOS allocates 30W without any CDP or LLDP negotiation. This means that even before the link comes up the class 4 power device gets 30W.

Also, on the hardware level the PSE does a 2-event classification which allows a class 4 PD to detect PSE capability of providing 30W from hardware, register itself and it can move up to PoE+ level without waiting for any CDP/LLDP packet exchange.

Once 2-event is enabled on a port, you need to manually shut/un-shut the port or connect the PD again to start the IEEE detection again. Power budget allocation for a class-4 device will be 30W if 2-event classification is enabled on the port, else it will be 15.4W.

Configuring 2-event Classification

To configure the switch for a 2-event Classification, perform the steps given below:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **power inline port 2-event**
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example:	Enables privileged EXEC mode. <ul style="list-style-type: none">• Enter your password if prompted.

	Command or Action	Purpose
	Device> enable	
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	interface <i>interface-id</i> Example: Device(config)# interface gigabitethernet2/0/1	Specifies the physical port to be configured, and enters interface configuration mode.
Step 4	power inline port 2-event Example: Device(config-if)# power inline port 2-event	Configures 2-event classification on the switch.
Step 5	end Example: Device(config-if)# end	Returns to privileged EXEC mode.

Example: Configuring 2-Event Classification

This example shows how you can configure 2-event classification.

```
Device> enable
Device# configure terminal
Device(config)# interface gigabitethernet2/0/1
Device(config-if)# power inline port 2-event
Device(config-if)# end
```




CHAPTER 10

Configuring EEE

- [Restrictions for EEE, on page 87](#)
- [Information About EEE, on page 87](#)
- [How to Configure EEE, on page 88](#)
- [Monitoring EEE, on page 89](#)
- [Configuration Examples for Configuring EEE, on page 90](#)
- [Additional References, on page 90](#)
- [Feature Information for Configuring EEE, on page 91](#)

Restrictions for EEE

EEE has the following restrictions:

- Changing the EEE configuration resets the interface because the device has to restart Layer 1 autonegotiation.
- You might want to enable the Link Layer Discovery Protocol (LLDP) for devices that require longer wakeup times before they are able to accept data on their receive paths. Doing so enables the device to negotiate for extended system wakeup times from the transmitting link partner.

Information About EEE

EEE Overview

Energy Efficient Ethernet (EEE) is an IEEE 802.3az standard that is designed to reduce power consumption in Ethernet networks during idle periods.

Default EEE Configuration

EEE is enabled by default.

How to Configure EEE

You can enable or disable EEE on an interface that is connected to an EEE-capable link partner.

Enabling or Disabling EEE

SUMMARY STEPS

1. **configure terminal**
2. **interface *interface-id***
3. **power efficient-ethernet auto**
4. **no power efficient-ethernet auto**
5. **end**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 2	interface <i>interface-id</i> Example: Device(config)# interface gigabitethernet 1/0/1	Specifies the interface to be configured, and enter interface configuration mode.
Step 3	power efficient-ethernet auto Example: Device(config-if)# power efficient-ethernet auto	Enables EEE on the specified interface. When EEE is enabled, the device advertises and autonegotiates EEE to its link partner.
Step 4	no power efficient-ethernet auto Example: Device(config-if)# no power efficient-ethernet auto	Disables EEE on the specified interface.
Step 5	end Example:	Returns to privileged EXEC mode.

	Command or Action	Purpose
	Device(config-if)# end	
Step 6	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Monitoring EEE

Table 11: Commands for Displaying EEE Settings

Command	Purpose
show eee capabilities interface <i>interface-id</i>	Displays EEE capabilities for the specified interface.
show eee status interface <i>interface-id</i>	Displays EEE status information for the specified interface.
show eee counters interface <i>interface-id</i>	Displays EEE counters for the specified interface.

Following are examples of the **show eee** commands

```
Switch#show eee capabilities interface gigabitEthernet2/0/1
Gi2/0/1
EEE(efficient-ethernet): yes (100-Tx and 1000T auto)
Link Partner : yes (100-Tx and 1000T auto)

ASIC/Interface : EEE Capable/EEE Enabled

Switch#show eee status interface gigabitEthernet2/0/1
Gi2/0/1 is up
EEE(efficient-ethernet): Operational
Rx LPI Status : Low Power
Tx LPI Status : Low Power
Wake Error Count : 0

ASIC EEE STATUS
Rx LPI Status : Receiving LPI
Tx LPI Status : Transmitting LPI
Link Fault Status : Link Up
Sync Status : Code group synchronization with data stream intact

Switch#show eee counters interface gigabitEthernet2/0/1

LP Active Tx Time (10us) : 66649648
LP Transitioning Tx : 462
LP Active Rx Time (10us) : 64911682
LP Transitioning Rx : 153
```

Examples for Cataylst Digital Building Series Switches

```
Switch#show eee capabilities interface gig1/0/1
Gi1/0/1
```

```

EEE(efficient-ethernet): yes (100-Tx and 1000T auto)
Link Partner : no

Switch#show eee status int gig1/0/1
Gig1/0/1 is up
EEE(efficient-ethernet): Disagreed
Rx LPI Status : None
Tx LPI Status : None
Wake Error Count : 0

```

Configuration Examples for Configuring EEE

This example shows how to enable EEE for an interface:

```

Device# configure terminal
Device(config)# interface gigabitethernet 1/0/1
Device(config-if)# power efficient-ethernet auto

```

This example shows how to disable EEE for an interface:

```

Device# configure terminal
Device(config)# interface gigabitethernet 1/0/1
Device(config-if)# no power efficient-ethernet auto

```

Additional References

MIBs

MIB	MIBs Link
All the supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
<p>The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies.</p> <p>To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds.</p> <p>Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.</p>	http://www.cisco.com/support

Feature Information for Configuring EEE

Release	Modification
Cisco IOS Release 15.0(2)EX1	This feature was introduced.



PART II

IP Multicast Routing

- [IP Multicast Routing Technology Overview, on page 95](#)
- [Configuring Basic IP Multicast Routing, on page 103](#)
- [Configuring IGMP, on page 117](#)
- [Configuring IGMP Snooping and Multicast VLAN Registration, on page 137](#)
- [Configuring CGMP, on page 183](#)
- [Configuring Protocol Independent Multicast \(PIM\), on page 189](#)
- [Configuring HSRP Aware PIM, on page 245](#)
- [Configuring VRRP Aware PIM, on page 253](#)
- [Configuring SSM, on page 259](#)



CHAPTER 11

IP Multicast Routing Technology Overview

- [Information About IP Multicast Technology, on page 95](#)
- [Additional References, on page 100](#)

Information About IP Multicast Technology

Role of IP Multicast in Information Delivery

IP multicast is a bandwidth-conserving technology that reduces traffic by delivering a single stream of information simultaneously to potentially thousands of businesses and homes. Applications that take advantage of multicast include video conferencing, corporate communications, distance learning, and distribution of software, stock quotes, and news.

IP multicast routing enables a host (source) to send packets to a group of hosts (receivers) anywhere within the IP network by using a special form of IP address called the IP multicast group address. The sending host inserts the multicast group address into the IP destination address field of the packet and IP multicast routers and multilayer switches forward incoming IP multicast packets out all interfaces that lead to the members of the multicast group. Any host, regardless of whether it is a member of a group, can send to a group. However, only the members of a group receive the message.

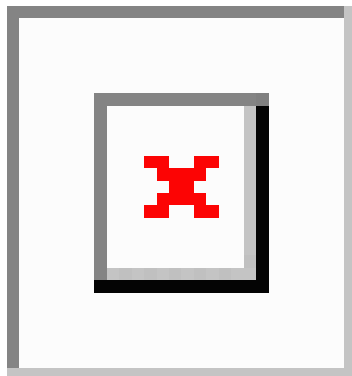
IP Multicast Routing Protocols

The software supports the following protocols to implement IP multicast routing:

- IGMP is used between hosts on a LAN and the routers (and multilayer devices) on that LAN to track the multicast groups of which hosts are members. To participate in IP multicasting, multicast hosts, routers, and multilayer devices must have the Internet Group Management Protocol (IGMP) operating.
- Protocol Independent Multicast (PIM) is used between routers so that they can track which multicast packets to forward to each other and to their directly connected LANs.
- IGMP Snooping is used for multicasting in a Layer 2 switching environment. It helps reduce the flooding of multicast traffic by dynamically configuring Layer 2 interfaces so that multicast traffic is forwarded to only those interfaces associated with IP multicast devices.

This figure shows where these protocols operate within the IP multicast environment.

Figure 5: IP Multicast Routing Protocols



According to IPv4 multicast standards, the MAC destination multicast address begins with 0100:5e and is appended by the last 23 bits of the IP address. For example, if the IP destination address is 239.1.1.39, the MAC destination address is 0100:5e01:0127.

A multicast packet is unmatched when the destination IPv4 address does not match the destination MAC address. The device forwards the unmatched packet in hardware based upon the MAC address table. If the destination MAC address is not in the MAC address table, the device floods the packet to the all port in the same VLAN as the receiving port.

Multicast Group Transmission Scheme

IP communication consists of hosts that act as senders and receivers of traffic as shown in the first figure. Senders are called sources. Traditional IP communication is accomplished by a single host source sending packets to another single host (unicast transmission) or to all hosts (broadcast transmission). IP multicast provides a third scheme, allowing a host to send packets to a subset of all hosts (multicast transmission). This subset of receiving hosts is called a multicast group. The hosts that belong to a multicast group are called group members.

Multicast is based on this group concept. A multicast group is an arbitrary number of receivers that join a group in order to receive a particular data stream. This multicast group has no physical or geographical boundaries--the hosts can be located anywhere on the Internet or on any private internetwork. Hosts that are interested in receiving data from a source to a particular group must join that group. Joining a group is accomplished by a host receiver by way of the Internet Group Management Protocol (IGMP).

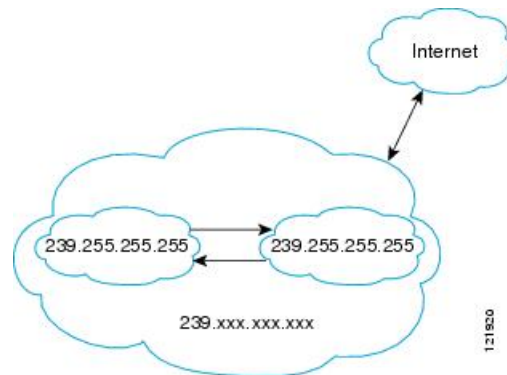
In a multicast environment, any host, regardless of whether it is a member of a group, can send to a group. However, only the members of a group can receive packets sent to that group. Multicast packets are delivered to a group using best-effort reliability, just like IP unicast packets.

In the next figure, the receivers (the designated multicast group) are interested in receiving the video data stream from the source. The receivers indicate their interest by sending an IGMP host report to the routers in the network. The routers are then responsible for delivering the data from the source to the receivers. The routers use Protocol Independent Multicast (PIM) to dynamically create a multicast distribution tree. The video data stream will then be delivered only to the network segments that are in the path between the source and the receivers.

IP Multicast Boundary

As shown in the figure, address scoping defines domain boundaries so that domains with RPs that have the same IP address do not leak into each other. Scoping is performed on the subnet boundaries within large domains and on the boundaries between the domain and the Internet.

Figure 6: Address Scoping at Boundaries



You can set up an administratively scoped boundary on an interface for multicast group addresses using the **ip multicast boundary** command with the *access-list* argument. A standard access list defines the range of addresses affected. When a boundary is set up, no multicast data packets are allowed to flow across the boundary from either direction. The boundary allows the same multicast group address to be reused in different administrative domains.

The Internet Assigned Numbers Authority (IANA) has designated the multicast address range 239.0.0.0 to 239.255.255.255 as the administratively scoped addresses. This range of addresses can be reused in domains administered by different organizations. They would be considered local, not globally unique.

You can configure the **filter-autorp** keyword to examine and filter Auto-RP discovery and announcement messages at the administratively scoped boundary. Any Auto-RP group range announcements from the Auto-RP packets that are denied by the boundary access control list (ACL) are removed. An Auto-RP group range announcement is permitted and passed by the boundary only if all addresses in the Auto-RP group range are permitted by the boundary ACL. If any address is not permitted, the entire group range is filtered and removed from the Auto-RP message before the Auto-RP message is forwarded.

IP Multicast Group Addressing

A multicast group is identified by its multicast group address. Multicast packets are delivered to that multicast group address. Unlike unicast addresses that uniquely identify a single host, multicast IP addresses do not identify a particular host. To receive the data sent to a multicast address, a host must join the group that address identifies. The data is sent to the multicast address and received by all the hosts that have joined the group indicating that they wish to receive traffic sent to that group. The multicast group address is assigned to a group at the source. Network administrators who assign multicast group addresses must make sure the addresses conform to the multicast address range assignments reserved by the Internet Assigned Numbers Authority (IANA).

IP Class D Addresses

IP multicast addresses have been assigned to the IPv4 Class D address space by IANA. The high-order four bits of a Class D address are 1110. Therefore, host group addresses can be in the range 224.0.0.0 to 239.255.255.255. A multicast address is chosen at the source (sender) for the receivers in a multicast group.



Note The Class D address range is used only for the group address or destination address of IP multicast traffic. The source address for multicast datagrams is always the unicast source address.

IP Multicast Address Scoping

The multicast address range is subdivided to provide predictable behavior for various address ranges and for address reuse within smaller domains. The table provides a summary of the multicast address ranges. A brief summary description of each range follows.

Table 12: Multicast Address Range Assignments

Name	Range	Description
Reserved Link-Local Addresses	224.0.0.0 to 224.0.0.255	Reserved for use by network protocols on a local network segment.
Globally Scoped Addresses	224.0.1.0 to 238.255.255.255	Reserved to send multicast data between organizations and across the Internet.
Source Specific Multicast	232.0.0.0 to 232.255.255.255	Reserved for use with the SSM datagram delivery model where data is forwarded only to receivers that have explicitly joined the group.
GLOP Addresses	233.0.0.0 to 233.255.255.255	Reserved for statically defined addresses by organizations that already have an assigned autonomous system (AS) domain number.
Limited Scope Address	239.0.0.0 to 239.255.255.255	Reserved as administratively or limited scope addresses for use in private multicast domains.

Reserved Link-Local Addresses

The IANA has reserved the range 224.0.0.0 to 224.0.0.255 for use by network protocols on a local network segment. Packets with an address in this range are local in scope and are not forwarded by IP routers. Packets with link local destination addresses are typically sent with a time-to-live (TTL) value of 1 and are not forwarded by a router.

Within this range, reserved link-local addresses provide network protocol functions for which they are reserved. Network protocols use these addresses for automatic router discovery and to communicate important routing information. For example, Open Shortest Path First (OSPF) uses the IP addresses 224.0.0.5 and 224.0.0.6 to exchange link-state information.

IANA assigns single multicast address requests for network protocols or network applications out of the 224.0.1.xxx address range. Multicast routers forward these multicast addresses.



Note All the packets with reserved link-local addresses are punted to CPU by default in the ASR 903 RSP2 Module.

Globally Scoped Addresses

Addresses in the range 224.0.1.0 to 238.255.255.255 are called globally scoped addresses. These addresses are used to send multicast data between organizations across the Internet. Some of these addresses have been reserved by IANA for use by multicast applications. For example, the IP address 224.0.1.1 is reserved for Network Time Protocol (NTP).

Source Specific Multicast Addresses

Addresses in the range 232.0.0.0/8 are reserved for Source Specific Multicast (SSM) by IANA. In Cisco IOS software, you can use the **ip pim ssm** command to configure SSM for arbitrary IP multicast addresses also. SSM is an extension of Protocol Independent Multicast (PIM) that allows for an efficient data delivery mechanism in one-to-many communications. SSM is described in the [IP Multicast Delivery Modes, on page 100](#) section.

GLOP Addresses

GLOP addressing (as proposed by RFC 2770, GLOP Addressing in 233/8) proposes that the 233.0.0.0/8 range be reserved for statically defined addresses by organizations that already have an AS number reserved. This practice is called GLOP addressing. The AS number of the domain is embedded into the second and third octets of the 233.0.0.0/8 address range. For example, AS 62010 is written in hexadecimal format as F23A. Separating the two octets F2 and 3A results in 242 and 58 in decimal format. These values result in a subnet of 233.242.58.0/24 that would be globally reserved for AS 62010 to use.

Limited Scope Addresses

The range 239.0.0.0 to 239.255.255.255 is reserved as administratively or limited scoped addresses for use in private multicast domains. These addresses are constrained to a local group or organization. Companies, universities, and other organizations can use limited scope addresses to have local multicast applications that will not be forwarded outside their domain. Routers typically are configured with filters to prevent multicast traffic in this address range from flowing outside an autonomous system (AS) or any user-defined domain. Within an AS or domain, the limited scope address range can be further subdivided so that local multicast boundaries can be defined.



Note Network administrators may use multicast addresses in this range, inside a domain, without conflicting with others elsewhere in the Internet.

Layer 2 Multicast Addresses

Historically, network interface cards (NICs) on a LAN segment could receive only packets destined for their burned-in MAC address or the broadcast MAC address. In IP multicast, several hosts need to be able to receive a single data stream with a common destination MAC address. Some means had to be devised so that multiple hosts could receive the same packet and still be able to differentiate between several multicast groups. One

method to accomplish this is to map IP multicast Class D addresses directly to a MAC address. Using this method, NICs can receive packets destined to many different MAC address.

Cisco Group Management Protocol (CGMP) is used on routers connected to Catalyst switches to perform tasks similar to those performed by IGMP. CGMP is necessary for those Catalyst switches that cannot distinguish between IP multicast data packets and IGMP report messages, both of which are addressed to the same group address at the MAC level.

IP Multicast Delivery Modes

IP multicast delivery modes differ only for the receiver hosts, not for the source hosts. A source host sends IP multicast packets with its own IP address as the IP source address of the packet and a group address as the IP destination address of the packet.

Source Specific Multicast

Source Specific Multicast (SSM) is a datagram delivery model that best supports one-to-many applications, also known as broadcast applications. SSM is a core network technology for the Cisco implementation of IP multicast targeted for audio and video broadcast application environments.

For the SSM delivery mode, an IP multicast receiver host must use IGMP Version 3 (IGMPv3) to subscribe to channel (S,G). By subscribing to this channel, the receiver host is indicating that it wants to receive IP multicast traffic sent by source host S to group G. The network will deliver IP multicast packets from source host S to group G to all hosts in the network that have subscribed to the channel (S, G).

SSM does not require group address allocation within the network, only within each source host. Different applications running on the same source host must use different SSM groups. Different applications running on different source hosts can arbitrarily reuse SSM group addresses without causing any excess traffic on the network.

Additional References

Related Documents

Related Topic	Document Title
IP multicast commands: complete command syntax, command mode, command history, defaults, usage guidelines and examples	<i>Cisco IOS IP Multicast Command Reference</i>
For complete syntax and usage information for the commands used in this chapter.	<i>IP Multicast Command Reference, Cisco IOS Release 15.2(2)E (Catalyst 2960-XR Switch)</i>

MIBs

MIB	MIBs Link
--	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

RFCs

RFC	Title
RFC 1112	Host Extensions for IP Multicasting
RFC 2113	IP Router Alert Option
RFC 2362	Protocol Independent Multicast-Sparse Mode (PIM-SM): Protocol Specification
RFC 3180	GLOP Addressing in 233/8

Technical Assistance

Description	Link
<p>The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies.</p> <p>To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds.</p> <p>Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.</p>	http://www.cisco.com/cisco/web/support/index.html



CHAPTER 12

Configuring Basic IP Multicast Routing

- [Prerequisites for Basic IP Multicast Routing, on page 103](#)
- [Restrictions for Basic IP Multicast Routing, on page 103](#)
- [Information About Basic IP Multicast Routing, on page 104](#)
- [How to Configure Basic IP Multicast Routing, on page 105](#)
- [Monitoring and Maintaining Basic IP Multicast Routing, on page 112](#)
- [Additional References, on page 115](#)
- [Feature History and Information for IP Multicast, on page 116](#)

Prerequisites for Basic IP Multicast Routing

The following are the prerequisites for configuring basic IP multicast routing:

- You must configure the PIM version and the PIM mode in order to perform IP multicast routing. The switch populates its multicast routing table and forwards multicast packets it receives from its directly connected LANs according to the mode setting. You can configure an interface to be in the PIM dense mode, sparse mode, or sparse-dense mode.
- Enabling PIM on an interface also enables IGMP operation on that interface. (To participate in IP multicasting, the multicast hosts, routers, and multilayer device must have IGMP operating.)

On a device running the IP base image, if you try to configure a VLAN interface with PIM dense-mode, sparse-mode, or dense-sparse-mode, the configuration is not allowed.

If you enable PIM on multiple interfaces, when most of these interfaces are not on the outgoing interface list, and IGMP snooping is disabled, the outgoing interface might not be able to sustain line rate for multicast traffic because of the extra replication.

Restrictions for Basic IP Multicast Routing

The following are the restrictions for IP multicast routing:

- The switch supports homogeneous stacking, but does not support mixed stacking.

Information About Basic IP Multicast Routing

IP multicasting is an efficient way to use network resources, especially for bandwidth-intensive services such as audio and video. IP multicast routing enables a host (source) to send packets to a group of hosts (receivers) anywhere within the IP network by using a special form of IP address called the IP multicast group address.

The sending host inserts the multicast group address into the IP destination address field of the packet, and IP multicast routers and multilayer devices forward incoming IP multicast packets out all interfaces that lead to members of the multicast group. Any host, regardless of whether it is a member of a group, can send to a group. However, only the members of a group receive the message.

Multicast Routing and Device Stacks

For all multicast routing protocols, the entire stack appears as a single router to the network and operates as a single multicast router.

In a device stack, the active device performs these functions:

- It is responsible for completing the IP multicast routing functions of the stack. It fully initializes and runs the IP multicast routing protocols.
- It builds and maintains the multicast routing table for the entire stack.
- It is responsible for distributing the multicast routing table to all stack members.

The stack members perform these functions:

- They act as multicast routing standby devices and are ready to take over if there is a active device failure. If the active device fails, all stack members delete their multicast routing tables. The newly elected active device starts building the routing tables and distributes them to the stack members.
- They do not build multicast routing tables. Instead, they use the multicast routing table that is distributed by the active device.

Default IP Multicast Routing Configuration

This table displays the default IP multicast routing configuration.

Table 13: Default IP Multicast Routing Configuration

Feature	Default Setting
Multicast routing	Disabled on all interfaces.
PIM version	Version 2.
PIM mode	No mode is defined.
PIM stub routing	None configured.
PIM RP address	None configured.

Feature	Default Setting
PIM domain border	Disabled.
PIM multicast boundary	None.
Candidate BSRs	Disabled.
Candidate RPs	Disabled.
Shortest-path tree threshold rate	0 kb/s.
PIM router query message interval	30 seconds.

Configuring sdr Listener Support

The MBONE is the small subset of Internet routers and hosts that are interconnected and capable of forwarding IP multicast traffic. Other multimedia content is often broadcast over the MBONE. Before you can join a multimedia session, you need to know what multicast group address and port are being used for the session, when the session is going to be active, and what sort of applications (audio, video, and so forth) are required on your workstation. The MBONE Session Directory Version 2 (sdr) tool provides this information. This freeware application can be downloaded from several sites on the World Wide Web, one of which is <http://www.video.ja.net/mice/index.html>.

SDR is a multicast application that listens to a well-known multicast group address and port for Session Announcement Protocol (SAP) multicast packets from SAP clients, which announce their conference sessions. These SAP packets contain a session description, the time the session is active, its IP multicast group addresses, media format, contact person, and other information about the advertised multimedia session. The information in the SAP packet is displayed in the SDR Session Announcement window.

How to Configure Basic IP Multicast Routing

Configuring Basic IP Multicast Routing

By default, multicast routing is disabled, and there is no default mode setting.

This procedure is required.

Before you begin

You must configure the PIM version and the PIM mode. The switch populates its multicast routing table and forwards multicast packets it receives from its directly connected LANs according to the mode setting.

In populating the multicast routing table, dense-mode interfaces are always added to the table. Sparse-mode interfaces are added to the table only when periodic join messages are received from downstream devices or when there is a directly connected member on the interface. When forwarding from a LAN, sparse-mode operation occurs if there is an RP known for the group. If so, the packets are encapsulated and sent toward the RP. When no RP is known, the packet is flooded in a dense-mode fashion. If the multicast traffic from a specific source is sufficient, the receiver's first-hop router might send join messages toward the source to build a source-based distribution tree.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip multicast-routing distributed**
4. **interface** *interface-id*
5. **ip pim version** [1 | 2]
6. **ip pim** {dense-mode | sparse-mode | sparse-dense-mode}
7. **end**
8. **show running-config**
9. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. Enter your password, if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	ip multicast-routing distributed Example: Device(config)# ip multicast-routing distributed	Enables IP multicast distributed switching Note To disable multicasting, use the no ip multicast-routing distributed global configuration command.
Step 4	interface <i>interface-id</i> Example: Device(config)# interface gigabitethernet 1/0/1	Specifies the Layer 3 interface on which you want to enable multicast routing, and enters interface configuration mode. The specified interface must be one of the following: <ul style="list-style-type: none"> • A routed port—A physical port that has been configured as a Layer 3 port by entering the no switchport interface configuration command. You will also need to enable IP PIM sparse-dense-mode on the interface, and join the interface as a statically connected member to an IGMP static group. • An SVI—A VLAN interface created by using the interface vlan <i>vlan-id</i> global configuration command. You will also need to enable IP PIM sparse-dense-mode on the VLAN, join the VLAN as a statically connected member to an IGMP static group,

	Command or Action	Purpose
		<p>and then enable IGMP snooping on the VLAN, the IGMP static group, and physical interface.</p> <p>These interfaces must have IP addresses assigned to them.</p>
Step 5	<p>ip pim version [1 2]</p> <p>Example:</p> <pre>Device(config-if)# ip pim version 2</pre>	<p>Configures the PIM version on the interface.</p> <p>By default, Version 2 is enabled and is the recommended setting.</p> <p>An interface in PIMv2 mode automatically downgrades to PIMv1 mode if that interface has a PIMv1 neighbor. The interface returns to Version 2 mode after all Version 1 neighbors are shut down or upgraded.</p> <p>Note To return to the default PIM version, use the no ip pim version interface configuration command.</p>
Step 6	<p>ip pim {dense-mode sparse-mode sparse-dense-mode}</p> <p>Example:</p> <pre>Device(config-if)# ip pim sparse-dense-mode</pre>	<p>Enables a PIM mode on the interface.</p> <p>By default, no mode is configured.</p> <p>The keywords have these meanings:</p> <ul style="list-style-type: none"> • dense-mode—Enables dense mode of operation. • sparse-mode—Enables sparse mode of operation. If you configure sparse mode, you must also configure an RP. • sparse-dense-mode—Causes the interface to be treated in the mode in which the group belongs. Sparse-dense mode is the recommended setting. <p>Note To disable PIM on an interface, use the no ip pim interface configuration command.</p>
Step 7	<p>end</p> <p>Example:</p> <pre>Device(config-if)# end</pre>	<p>Returns to privileged EXEC mode.</p>
Step 8	<p>show running-config</p> <p>Example:</p> <pre>Device# show running-config</pre>	<p>Verifies your entries.</p>
Step 9	<p>copy running-config startup-config</p> <p>Example:</p>	<p>(Optional) Saves your entries in the configuration file.</p>

	Command or Action	Purpose
	Device# <code>copy running-config startup-config</code>	

Configuring Optional IP Multicast Routing Features

Defining the IP Multicast Boundary

You define a multicast boundary to prevent Auto-RP messages from entering the PIM domain. You create an access list to deny packets destined for 224.0.1.39 and 224.0.1.40, which carry Auto-RP information.

This procedure is optional.

SUMMARY STEPS

1. `enable`
2. `configure terminal`
3. `access-list access-list-number deny source [source-wildcard]`
4. `interface interface-id`
5. `ip multicast boundary access-list-number`
6. `end`
7. `show running-config`
8. `copy running-config startup-config`

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p><code>enable</code></p> <p>Example:</p> <pre>Device> enable</pre>	<p>Enables privileged EXEC mode.</p> <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	<p><code>configure terminal</code></p> <p>Example:</p> <pre>Device# configure terminal</pre>	<p>Enters global configuration mode.</p>
Step 3	<p><code>access-list access-list-number deny source [source-wildcard]</code></p> <p>Example:</p> <pre>Device(config)# access-list 12 deny 224.0.1.39 access-list 12 deny 224.0.1.40</pre>	<p>Creates a standard access list, repeating the command as many times as necessary.</p> <ul style="list-style-type: none"> • For <i>access-list-number</i>, the range is 1 to 99. • The deny keyword denies access if the conditions are matched. • For <i>source</i>, enter multicast addresses 224.0.1.39 and 224.0.1.40, which carry Auto-RP information.

	Command or Action	Purpose
		<ul style="list-style-type: none"> (Optional) For <i>source-wildcard</i>, enter the wildcard bits in dotted decimal notation to be applied to the source. Place ones in the bit positions that you want to ignore. <p>The access list is always terminated by an implicit deny statement for everything.</p>
Step 4	interface <i>interface-id</i> Example: Device(config)# interface gigabitethernet 1/0/1	Specifies the interface to be configured, and enters interface configuration mode.
Step 5	ip multicast boundary <i>access-list-number</i> Example: Device(config-if)# ip multicast boundary 12	Configures the boundary, specifying the access list you created in Step 2. Note To remove the boundary, use the no ip multicast boundary interface configuration command.
Step 6	end Example: Device(config)# end	Returns to privileged EXEC mode.
Step 7	show running-config Example: Device# show running-config	Verifies your entries.
Step 8	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Configuring Multicast VRFs

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters global configuration mode.

	Command or Action	Purpose
Step 2	ip routing Example: Device(config)# ip routing	Enables IP routing mode.
Step 3	ip vrf vrf-name Example: Device(config)# ip vrf vpn1	Names the VRF, and enter VRF configuration mode.
Step 4	rd route-distinguisher Example: Device(config-vrf)# rd 100:2	Creates a VRF table by specifying a route distinguisher. Enter either an AS number and an arbitrary number (xxx:y) or an IP address and an arbitrary number (A.B.C.D:y)
Step 5	route-target {export import both} <i>route-target-ext-community</i> Example: Device(config-vrf)# route-target import 100:2	Creates a list of import, export, or import and export route target communities for the specified VRF. Enter either an AS system number and an arbitrary number (xxx:y) or an IP address and an arbitrary number (A.B.C.D:y). The <i>route-target-ext-community</i> should be the same as the <i>route-distinguisher</i> entered in Step 4.
Step 6	import map route-map Example: Device(config-vrf)# import map importmap1	(Optional) Associates a route map with the VRF.
Step 7	ip multicast-routing vrf vrf-name distributed Example: Device(config-vrf)# ip multicast-routing vrf vpn1 distributed	(Optional) Enables global multicast routing for VRF table.
Step 8	interface interface-id Example: Device(config-vrf)# interface gigabitethernet 1/0/2	Specifies the Layer 3 interface to be associated with the VRF, and enter interface configuration mode. The interface can be a routed port or an SVI.
Step 9	ip vrf forwarding vrf-name Example: Device(config-if)# ip vrf forwarding vpn1	Associates the VRF with the Layer 3 interface.
Step 10	ip address ip-address mask Example:	Configures IP address for the Layer 3 interface.

	Command or Action	Purpose
	Device(config-if)# ip address 10.1.5.1 255.255.255.0	
Step 11	ip pim sparse-dense mode Example: Device(config-if)# ip pim sparse-dense mode	Enables PIM on the VRF-associated Layer 3 interface.
Step 12	end Example: Device(config)# end	Returns to privileged EXEC mode.
Step 13	show ip vrf [brief detail interfaces] [vrf-name] Example: Device# show ip vrf detail vpn1	Verifies the configuration. Displays information about the configured VRFs.
Step 14	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Advertising Multicast Multimedia Sessions Using SAP Listener

Enable SAP listener support when you want to use session description and announcement protocols and applications to assist the advertisement of multicast multimedia conferences and other multicast sessions and to communicate the relevant session setup information to prospective participants.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip sap cache-timeout** *minutes*
4. **interface** *type number*
5. **ip sap listen**
6. **end**
7. **clear ip sap** [*group-address* | “*session-name*”]
8. **show ip sap** [*group-address* | “*session-name*”] **detail**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable	Enables privileged EXEC mode.

	Command or Action	Purpose
	Example: Router> enable	<ul style="list-style-type: none"> Enter your password if prompted.
Step 2	configure terminal Example: Router# configure terminal	Enters global configuration mode.
Step 3	ip sap cache-timeout <i>minutes</i> Example: Router(config)# ip sap cache-timeout 600	(Optional) Limits how long a SAP cache entry stays active in the cache. <ul style="list-style-type: none"> By default, SAP cache entries are deleted 24 hours after they are received from the network.
Step 4	interface <i>type number</i> Example: Router(config)# interface ethernet 1	Selects an interface that is connected to hosts on which IGMPv3 can be enabled.
Step 5	ip sap listen Example: Router(config-if)# ip sap listen	Enables the software to listen to session directory announcements.
Step 6	end Example: Router(config-if)# end	Ends the session and returns to EXEC mode.
Step 7	clear ip sap [<i>group-address</i> “<i>session-name</i>”] Example: Router# clear ip sap "Sample Session"	Deletes a SAP cache entry or the entire SAP cache.
Step 8	show ip sap [<i>group-address</i> “<i>session-name</i>”] detail] Example: Router# show ip sap 224.2.197.250 detail	(Optional) Displays the SAP cache.

Monitoring and Maintaining Basic IP Multicast Routing

Clearing Caches, Tables, and Databases

You can remove all contents of a particular cache, table, or database. Clearing a cache, table, or database might be necessary when the contents of the particular structure are or suspected to be invalid.

You can use any of the privileged EXEC commands in the following table to clear IP multicast caches, tables, and databases.

Table 14: Commands for Clearing Caches, Tables, and Databases

Command	Purpose
clear ip cgmp	Clears all group entries the Catalyst switches have cached.
clear ip igmp group { group [<i>hostname</i> <i>IP address</i>] vrf name group [<i>hostname</i> <i>IP address</i>] }	Deletes entries from the IGMP cache.
clear ip mroute { * [<i>hostname</i> <i>IP address</i>] vrf name group [<i>hostname</i> <i>IP address</i>] }	Deletes entries from the IP multicast routing table.
clear ip sap [<i>group-address</i> “ <i>session-name</i> ”]	Deletes the Session Directory Protocol Version 2 cache or an sdr cache entry.

Displaying System and Network Statistics

You can display specific statistics, such as the contents of IP routing tables, caches, and databases.



Note This release does not support per-route statistics.

You can display information to learn resource usage and solve network problems. You can also display information about node reachability and discover the routing path that packets of your device are taking through the network.

You can use any of the privileged EXEC commands in the following table to display various routing statistics.

Table 15: Commands for Displaying System and Network Statistics

Command	Purpose
ping [<i>group-name</i> <i>group-address</i>]	Sends an ICMP Echo Request to a multicast group address.
show ip igmp groups [<i>group-name</i> <i>group-address</i> <i>type-number</i>]	Displays the multicast groups that are directly connected to the device and that were learned through IGMP.
show ip igmp interface [<i>type number</i>]	Displays multicast-related information about an interface.
show ip mroute [<i>group-name</i> <i>group-address</i>] [<i>source</i>] [count interface proxy pruned summary verbose]	Displays the contents of the IP multicast routing table.

Command	Purpose
show ip pim interface [<i>type number</i>] [count detail df stats]	Displays information about interfaces configured for PIM. This command is available in all software images.
show ip pim neighbor [<i>type number</i>]	Lists the PIM neighbors discovered by the device. This command is available in all software images.
show ip pim rp [<i>group-name</i> <i>group-address</i>]	Displays the RP routers associated with a sparse-mode multicast group. This command is available in all software images.
show ip rpf { <i>source-address</i> <i>name</i> }	Displays how the device is doing Reverse-Path Forwarding (that is, from the unicast routing table, DVMRP routing table, or static mroutes). Command parameters include: <ul style="list-style-type: none"> • <i>Host name</i> or <i>IP address</i>—IP name or group address. • Select—Group-based VRF select information. • vrf—Selects VPN Routing/Forwarding instance.
show ip sap [<i>group</i> " <i>session-name</i> " detail]	Displays the Session Announcement Protocol (SAP) Version 2 cache. Command parameters include: <ul style="list-style-type: none"> • <i>A.B.C.D</i>—IP group address. • <i>WORD</i>—Session name (in double quotes). • detail—Session details.

Displaying Multicast Peers, Packet Rates and Loss Information, and Path Tracing

You can use the privileged EXEC commands in the following table to monitor IP multicast routers, packets, and paths.

Table 16: Commands for Displaying Multicast Peers, Packet Rates and Loss Information, and Path Tracing

Command	Purpose
mrinfo { [<i>hostname</i> <i>address</i>] vrf }	Queries a multicast router or multilayer device about which neighboring multicast devices are peering with it.
mstat { [<i>hostname</i> <i>address</i>] vrf }	Displays IP multicast packet rate and information loss.

Additional References

Related Documents

Related Topic	Document Title
For complete syntax and usage information for the commands used in this chapter.	<i>IP Multicast Command Reference, Cisco IOS Release 15.2(2)E (Catalyst 2960-XR Switch)</i>
Cisco IOS IP multicast commands	Cisco IOS IP Multicast Command Reference

Standards and RFCs

Standard/RFC	Title
RFC 1112	<i>Host Extensions for IP Multicasting</i>
RFC 2236	<i>Internet Group Management Protocol, Version 2</i>
RFC 4601	<i>Protocol-Independent Multicast-Sparse Mode (PIM-SM): Protocol Specification</i>

MIBs

MIB	MIBs Link
All the supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
<p>The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies.</p> <p>To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds.</p> <p>Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.</p>	http://www.cisco.com/support

Feature History and Information for IP Multicast

Release	Modification
Cisco IOS Release 15.0(2)EX1	This feature was introduced.



CHAPTER 13

Configuring IGMP

- [Prerequisites for IGMP, on page 117](#)
- [Restrictions for Configuring IGMP, on page 117](#)
- [Information About IGMP, on page 118](#)
- [How to Configure IGMP, on page 123](#)
- [Monitoring IGMP, on page 134](#)
- [Configuration Examples for IGMP, on page 135](#)
- [Additional References, on page 135](#)
- [Feature History and Information for IGMP, on page 136](#)

Prerequisites for IGMP

- Before performing the tasks in this module, you should be familiar with the concepts explained in the "IP Multicast Routing Technology Overview" module.
- The tasks in this module assume that IP multicast has been enabled and that the Protocol Independent Multicast (PIM) interfaces have been configured using the tasks described in the "Configuring Basic IP Multicast Routing" module.

Restrictions for Configuring IGMP

The following are the restrictions for configuring IGMP:

- The device supports IGMP Versions 1, 2, and 3.



Note For IGMP Version 3, only IGMP Version 3 BISS (Basic IGMPv3 Snooping Support) is supported.

- IGMP Version 3 uses new membership report messages that might not be correctly recognized by older IGMP snooping devices.
- IGMPv3 can operate with both ISM and SSM. In ISM, both exclude and include mode reports are applicable. In SSM, only include mode reports are accepted by the last-hop router. Exclude mode reports are ignored.

Information About IGMP

Role of the Internet Group Management Protocol

IGMP is used to dynamically register individual hosts in a multicast group on a particular LAN. Enabling PIM on an interface also enables IGMP. IGMP provides a means to automatically control and limit the flow of multicast traffic throughout your network with the use of special multicast queriers and hosts.

- A querier is a network device, such as a router, that sends query messages to discover which network devices are members of a given multicast group.
- A host is a receiver, including routers, that sends report messages (in response to query messages) to inform the querier of a host membership. Hosts use IGMP messages to join and leave multicast groups.

Hosts identify group memberships by sending IGMP messages to their local multicast device. Under IGMP, devices listen to IGMP messages and periodically send out queries to discover which groups are active or inactive on a particular subnet.

IGMP Multicast Addresses

IP multicast traffic uses group addresses, which are Class D IP addresses. The high-order four bits of a Class D address are 1110. Therefore, host group addresses can be in the range 224.0.0.0 to 239.255.255.255.

Multicast addresses in the range 224.0.0.0 to 224.0.0.255 are reserved for use by routing protocols and other network control traffic. The address 224.0.0.0 is guaranteed not to be assigned to any group.

IGMP packets are transmitted using IP multicast group addresses as follows:

- IGMP general queries are destined to the address 224.0.0.1 (all systems on a subnet).
- IGMP group-specific queries are destined to the group IP address for which the device is querying.
- IGMP group membership reports are destined to the group IP address for which the device is reporting.
- IGMPv2 leave-group messages are destined to the address 224.0.0.2 (all devices on a subnet).
- IGMPv3 membership reports are destined to the address 224.0.0.22; all IGMPv3-capable multicast devices must listen to this address.

IGMP Versions

The device supports IGMP version 1, IGMP version 2, and IGMP version 3. These versions are interoperable on the device. For example, if IGMP snooping is enabled and the querier's version is IGMPv2, and the device receives an IGMPv3 report from a host, then the device can forward the IGMPv3 report to the multicast router.

An IGMPv3 device can receive messages from and forward messages to a device running the Source Specific Multicast (SSM) feature.

IGMP Version 1

IGMP version 1 (IGMPv1) primarily uses a query-response model that enables the multicast router and multilayer device to find which multicast groups are active (have one or more hosts interested in a multicast group) on the local subnet. IGMPv1 has other processes that enable a host to join and leave a multicast group. For more information, see RFC 1112.

IGMP Version 2

IGMPv2 extends IGMP functionality by providing such features as the IGMP leave process to reduce leave latency, group-specific queries, and an explicit maximum query response time. IGMPv2 also adds the capability for routers to elect the IGMP querier without depending on the multicast protocol to perform this task. For more information, see RFC 2236.



Note IGMP version 2 is the default version for the device.

IGMP Version 3

The device supports IGMP version 3.

An IGMPv3 device supports Basic IGMPv3 Snooping Support (BISS), which includes support for the snooping features on IGMPv1 and IGMPv2 switches and for IGMPv3 membership report messages. BISS constrains the flooding of multicast traffic when your network includes IGMPv3 hosts. It constrains traffic to approximately the same set of ports as the IGMP snooping feature on IGMPv2 or IGMPv1 hosts.

An IGMPv3 device can receive messages from and forward messages to a device running the Source Specific Multicast (SSM) feature.

IGMPv3 Host Signalling

In IGMPv3, hosts signal membership to last hop routers of multicast groups. Hosts can signal group membership with filtering capabilities with respect to sources. A host can either signal that it wants to receive traffic from all sources sending to a group except for some specific sources (called exclude mode), or that it wants to receive traffic only from some specific sources sending to the group (called include mode).

IGMPv3 can operate with both Internet Standard Multicast (ISM) and Source Specific Multicast (SSM). In ISM, both exclude and include mode reports are applicable. In SSM, only include mode reports are accepted by the last-hop router. Exclude mode reports are ignored.

IGMP Versions Differences

There are three versions of IGMP, as defined by Request for Comments (RFC) documents of the Internet Engineering Task Force (IETF). IGMPv2 improves over IGMPv1 by adding the ability for a host to signal desire to leave a multicast group and IGMPv3 improves over IGMPv2 mainly by adding the ability to listen to multicast originating from a set of source IP addresses only.

Table 17: IGMP Versions

IGMP Version	Description
IGMPv1	Provides the basic query-response mechanism that allows the multicast device to determine which multicast groups are active and other processes that enable hosts to join and leave a multicast group. RFC 1112 defines the IGMPv1 host extensions for IP multicasting.
IGMPv2	Extends IGMP, allowing such capabilities as the IGMP leave process, group-specific queries, and an explicit maximum response time field. IGMPv2 also adds the capability for devices to elect the IGMP querier without dependence on the multicast protocol to perform this task. RFC 2236 defines IGMPv2.



Note By default, enabling a PIM on an interface enables IGMPv2 on that device. IGMPv2 was designed to be as backward compatible with IGMPv1 as possible. To accomplish this backward compatibility, RFC 2236 defined special interoperability rules. If your network contains legacy IGMPv1 hosts, you should be familiar with these operability rules. For more information about IGMPv1 and IGMPv2 interoperability, see RFC 2236, Internet Group Management Protocol, Version 2 .

Devices That Run IGMPv1

IGMPv1 devices send IGMP queries to the “all-hosts” multicast address of 224.0.0.1 to solicit multicast groups with active multicast receivers. The multicast receivers also can send IGMP reports to the device to notify it that they are interested in receiving a particular multicast stream. Hosts can send the report asynchronously or in response to the IGMP queries sent by the device. If more than one multicast receiver exists for the same multicast group, only one of these hosts sends an IGMP report message; the other hosts suppress their report messages.

In IGMPv1, there is no election of an IGMP querier. If more than one device on the segment exists, all the devices send periodic IGMP queries. IGMPv1 has no special mechanism by which the hosts can leave the group. If the hosts are no longer interested in receiving multicast packets for a particular group, they simply do not reply to the IGMP query packets sent from the device. The device continues sending query packets. If the device does not hear a response in three IGMP queries, the group times out and the device stops sending multicast packets on the segment for the group. If the host later wants to receive multicast packets after the timeout period, the host simply sends a new IGMP join to the device, and the device begins to forward the multicast packet again.

If there are multiple devices on a LAN, a designated router (DR) must be elected to avoid duplicating multicast traffic for connected hosts. PIM devices follow an election process to select a DR. The PIM device with the highest IP address becomes the DR.

The DR is responsible for the following tasks:

- Sending PIM register and PIM Join and Prune messages toward the rendezvous point (RP) to inform it about host group membership.
- Sending IGMP host-query messages.
- Sending host-query messages by default every 60 seconds in order to keep the IGMP overhead on hosts and networks very low.

Devices That Run IGMPv2

IGMPv2 improves the query messaging capabilities of IGMPv1.

The query and membership report messages in IGMPv2 are identical to the IGMPv1 messages with two exceptions:

- IGMPv2 query messages are broken into two categories: general queries (identical to IGMPv1 queries) and group-specific queries.
- IGMPv1 membership reports and IGMPv2 membership reports have different IGMP type codes.

IGMPv2 also enhances IGMP by providing support for the following capabilities:

- Querier election process--Provides the capability for IGMPv2 devices to elect the IGMP querier without having to rely on the multicast routing protocol to perform the process.
- Maximum Response Time field--A new field in query messages permits the IGMP querier to specify the maximum query-response time. This field permits the tuning of the query-response process to control response burstiness and to fine-tune leave latencies.
- Group-Specific Query messages--Permits the IGMP querier to perform the query operation on a specific group instead of all groups.
- Leave-Group messages--Provides hosts with a method of notifying devices on the network that they wish to leave the group.

Unlike IGMPv1, in which the DR and the IGMP querier are typically the same device, in IGMPv2 the two functions are decoupled. The DR and the IGMP querier are selected based on different criteria and may be different devices on the same subnet. The DR is the device with the highest IP address on the subnet, whereas the IGMP querier is the device with the lowest IP address.

Query messages are used to elect the IGMP querier as follows:

1. When IGMPv2 devices start, they each multicast a general query message to the all-systems group address of 224.0.0.1 with their interface address in the source IP address field of the message.
2. When an IGMPv2 device receives a general query message, the device compares the source IP address in the message with its own interface address. The device with the lowest IP address on the subnet is elected the IGMP querier.
3. All devices (excluding the querier) start the query timer, which is reset whenever a general query message is received from the IGMP querier. If the query timer expires, it is assumed that the IGMP querier has gone down, and the election process is performed again to elect a new IGMP querier.

By default, the timer is two times the query interval.

IGMP Join and Leave Process

IGMP Join Process

When a host wants to join a multicast group, the host sends one or more unsolicited membership reports for the multicast group it wants to join. The IGMP join process is the same for IGMPv1 and IGMPv2 hosts.

In IGMPv3, the join process for hosts proceeds as follows:

- When a host wants to join a group, it sends an IGMPv3 membership report to 224.0.0.22 with an empty EXCLUDE list.
- When a host wants to join a specific channel, it sends an IGMPv3 membership report to 224.0.0.22 with the address of the specific source included in the INCLUDE list.
- When a host wants to join a group excluding particular sources, it sends an IGMPv3 membership report to 224.0.0.22 excluding those sources in the EXCLUDE list.



Note If some IGMPv3 hosts on a LAN wish to exclude a source and others wish to include the source, then the device will send traffic for the source on the LAN (that is, inclusion trumps exclusion in this situation).

IGMP Leave Process

The method that hosts use to leave a group varies depending on the version of IGMP in operation.

IGMPv1 Leave Process

There is no leave-group message in IGMPv1 to notify the devices on the subnet that a host no longer wants to receive the multicast traffic from a specific group. The host simply stops processing traffic for the multicast group and ceases responding to IGMP queries with IGMP membership reports for the group. As a result, the only way IGMPv1 devices know that there are no longer any active receivers for a particular multicast group on a subnet is when the devices stop receiving membership reports. To facilitate this process, IGMPv1 devices associate a countdown timer with an IGMP group on a subnet. When a membership report is received for the group on the subnet, the timer is reset. For IGMPv1 devices, this timeout interval is typically three times the query interval (3 minutes). This timeout interval means that the device may continue to forward multicast traffic onto the subnet for up to 3 minutes after all hosts have left the multicast group.

IGMPv2 Leave Process

IGMPv2 incorporates a leave-group message that provides the means for a host to indicate that it wishes to stop receiving multicast traffic for a specific group. When an IGMPv2 host leaves a multicast group, if it was the last host to respond to a query with a membership report for that group, it sends a leave-group message to the all-devices multicast group (224.0.0.2).

IGMPv3 Leave Process

IGMPv3 enhances the leave process by introducing the capability for a host to stop receiving traffic from a particular group, source, or channel in IGMP by including or excluding sources, groups, or channels in IGMPv3 membership reports.

Default IGMP Configuration

This table displays the default IGMP configuration for the device.

Table 18: Default IGMP Configuration

Feature	Default Setting
Multilayer device as a member of a multicast group	No group memberships are defined.

Feature	Default Setting
Access to multicast groups	All groups are allowed on an interface.
IGMP version	Version 2 on all interfaces.
IGMP host-query message interval	60 seconds on all interfaces.
IGMP query timeout	60 seconds on all interfaces.
IGMP maximum query response time	10 seconds on all interfaces.
Multilayer device as a statically connected member	Disabled.

How to Configure IGMP

Configuring the Device as a Member of a Group

You can configure the device as a member of a multicast group and discover multicast reachability in a network. If all the multicast-capable routers and multilayer devices that you administer are members of a multicast group, pinging that group causes all of these devices to respond. The devices respond to ICMP echo-request packets addressed to a group of which they are members. Another example is the multicast trace-route tools provided in the software.



Caution Performing this procedure might impact the CPU performance because the CPU will receive all data traffic for the group address.

This procedure is optional.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **ip igmp join-group** *group-address*
5. **end**
6. **show ip igmp interface** [*interface-id*]
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example:	Enables privileged EXEC mode. • Enter your password if prompted.

	Command or Action	Purpose
	Device> enable	
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	interface <i>interface-id</i> Example: Device(config)# interface gigabitethernet 1/0/1	Specifies the interface on which you want to enable multicast routing, and enters interface configuration mode.
Step 4	ip igmp join-group <i>group-address</i> Example: Device(config-if)# ip igmp join-group 225.2.2.2	Configures the device to join a multicast group. By default, no group memberships are defined. For <i>group-address</i> , specify the multicast IP address in dotted decimal notation. Note To cancel membership in a group, use the no ip igmp join-group <i>group-address</i> interface configuration command.
Step 5	end Example: Device(config)# end	Returns to privileged EXEC mode.
Step 6	show ip igmp interface [<i>interface-id</i>] Example: Device# show ip igmp interface	Verifies your entries.
Step 7	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Controlling Access to IP Multicast Group

The switch sends IGMP host-query messages to find which multicast groups have members on attached local networks. The switch then forwards to these group members all packets addressed to the multicast group.

You can place a filter on each interface to restrict the multicast groups that hosts on the subnet serviced by the interface can join.

This procedure is optional.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **ip igmp access-group** *access-list-number*
5. **exit**
6. **access-list** *access-list-number* {**deny** | **permit**} *source* [*source-wildcard*]
7. **end**
8. **show ip igmp interface** [*interface-id*]

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	interface <i>interface-id</i> Example: Device(config)# interface GigabitEthernet 1/0/12	Specifies the interface to be configured, and enters interface configuration mode.
Step 4	ip igmp access-group <i>access-list-number</i> Example: Device(config-if)# ip igmp access-group 10	Specifies the multicast groups that hosts on the subnet serviced by an interface can join. By default, all groups are allowed on an interface. For <i>access-list-number</i> , specify an IP standard access list number. The range is 1 to 199. Note To disable groups on an interface, use the no ip igmp access-group interface configuration command.

	Command or Action	Purpose
Step 5	exit Example: <pre>Device(config-if)# exit</pre>	Returns to global configuration mode.
Step 6	access-list <i>access-list-number</i> { deny permit } <i>source</i> [<i>source-wildcard</i>] Example: <pre>Device(config)# access-list 10 permit</pre>	Creates a standard access list. <ul style="list-style-type: none"> • For <i>access-list-number</i>, specify the access list created in Step 3. • The deny keyword denies access if the conditions are matched. The permit keyword permits access if the conditions are matched. • For <i>source</i>, specify the multicast group that hosts on the subnet can join. • (Optional) For <i>source-wildcard</i>, enter the wildcard bits in dotted decimal notation to be applied to the source. Place ones in the bit positions that you want to ignore. Recall that the access list is always terminated by an implicit deny statement for everything.
Step 7	end Example: <pre>Device(config-igmp-profile)# end</pre>	Returns to privileged EXEC mode.
Step 8	show ip igmp interface [<i>interface-id</i>] Example: <pre>Device# show ip igmp interface</pre>	Verifies your entries.

Changing the IGMP Version

By default, the switch uses IGMP Version 2, which provides features such as the IGMP query timeout and the maximum query response time.

All systems on the subnet must support the same version. The switch does not automatically detect Version 1 systems and switch to Version 1. You can mix Version 1 and Version 2 hosts on the subnet because Version 2 routers or switches always work correctly with IGMPv1 hosts.

Configure the switch for Version 1 if your hosts do not support Version 2.

This procedure is optional.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **ip igmp version** {1 | 2 | 3 }
5. **end**
6. **show ip igmp interface** [*interface-id*]
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Device> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: <pre>Device# configure terminal</pre>	Enters global configuration mode.
Step 3	interface <i>interface-id</i> Example: <pre>Device(config)# interface gigabitethernet 1/0/1</pre>	Specifies the interface to be configured, and enters the interface configuration mode.
Step 4	ip igmp version {1 2 3 } Example: <pre>Device(config-if)# ip igmp version 2</pre>	Specifies the IGMP version that the switch uses. Note If you change to Version 1, you cannot configure the ip igmp query-interval or the ip igmp query-max-response-time interface configuration commands. To return to the default setting, use the no ip igmp version interface configuration command.
Step 5	end Example: <pre>Device(config)# end</pre>	Returns to privileged EXEC mode.
Step 6	show ip igmp interface [<i>interface-id</i>] Example:	Verifies your entries.

	Command or Action	Purpose
	Device# <code>show ip igmp interface</code>	
Step 7	copy running-config startup-config Example: Device# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Modifying the IGMP Host-Query Message Interval

The device periodically sends IGMP host-query messages to discover which multicast groups are present on attached networks. These messages are sent to the all-hosts multicast group (224.0.0.1) with a time-to-live (TTL) of 1. The device sends host-query messages to refresh its knowledge of memberships present on the network. If, after some number of queries, the software discovers that no local hosts are members of a multicast group, the software stops forwarding multicast packets to the local network from remote origins for that group and sends a prune message upstream toward the source.

The device elects a PIM designated router (DR) for the LAN (subnet). The designated router is responsible for sending IGMP host-query messages to all hosts on the LAN. In sparse mode, the designated router also sends PIM register and PIM join messages toward the RP router. With IGMPv2, the DR is the router or multilayer device with the highest IP address. With IGMPv1, the DR is elected according to the multicast routing protocol that runs on the LAN.

This procedure is optional.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **ip igmp query-interval** *seconds*
5. **end**
6. **show ip igmp interface** [*interface-id*]
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> <code>enable</code>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example:	Enters global configuration mode.

	Command or Action	Purpose
	Device# <code>configure terminal</code>	
Step 3	interface <i>interface-id</i> Example: Device(config)# interface gigabitethernet 1/0/1	Specifies the interface on which you want to enable multicast routing, and enters interface configuration mode.
Step 4	ip igmp query-interval <i>seconds</i> Example: Device(config-if)# ip igmp query-interval 75	Configures the frequency at which the designated router sends IGMP host-query messages. By default, the designated router sends IGMP host-query messages every 60 seconds to keep the IGMP overhead very low on hosts and networks. The range is 1 to 18000. Note To return to the default setting, use the no ip igmp query-interval interface configuration command.
Step 5	end Example: Device(config)# end	Returns to privileged EXEC mode.
Step 6	show ip igmp interface [<i>interface-id</i>] Example: Device# show ip igmp interface	Verifies your entries.
Step 7	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Changing the IGMP Query Timeout for IGMPv2

If you are using IGMPv2, you can specify the period of time before the device takes over as the querier for the interface. By default, the device waits twice the query interval period controlled by the **ip igmp query-interval** interface configuration command. After that time, if the device has received no queries, it becomes the querier.

This procedure is optional.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **ip igmp querier-timeout** *seconds*
5. **end**
6. **show ip igmp interface** [*interface-id*]
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Device> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: <pre>Device# configure terminal</pre>	Enters global configuration mode.
Step 3	interface <i>interface-id</i> Example: <pre>Device(config)# interface gigabitethernet 1/0/1</pre>	Specifies the interface on which you want to enable multicast routing, and enters interface configuration mode.
Step 4	ip igmp querier-timeout <i>seconds</i> Example: <pre>Device(config-if)# ip igmp querier-timeout 120</pre>	Specifies the IGMP query timeout. The default is 60 seconds (twice the query interval). The range is 60 to 300. Note To return to the default setting, use the no ip igmp querier-timeout interface configuration command.
Step 5	end Example: <pre>Device(config)# end</pre>	Returns to privileged EXEC mode.
Step 6	show ip igmp interface [<i>interface-id</i>] Example:	Verifies your entries.

	Command or Action	Purpose
	Device# <code>show ip igmp interface</code>	
Step 7	copy running-config startup-config Example: Device# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Changing the Maximum Query Response Time for IGMPv2

If you are using IGMPv2, you can change the maximum query response time advertised in IGMP queries. The maximum query response time enables the device to quickly detect that there are no more directly connected group members on a LAN. Decreasing the value enables the device to prune groups faster.

This procedure is optional.

SUMMARY STEPS

1. `enable`
2. `configure terminal`
3. `interface interface-id`
4. `ip igmp query-max-response-time seconds`
5. `end`
6. `show ip igmp interface [interface-id]`
7. `copy running-config startup-config`

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> <code>enable</code>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode.
Step 3	interface interface-id Example: Device(config)# <code>interface</code>	Specifies the interface on which you want to enable multicast routing, and enters interface configuration mode.

	Command or Action	Purpose
	<code>gigabitethernet 1/0/1</code>	
Step 4	<p>ip igmp query-max-response-time <i>seconds</i></p> <p>Example:</p> <pre>Device(config-if)# ip igmp query-max-response-time 15</pre>	<p>Changes the maximum query response time advertised in IGMP queries.</p> <p>The default is 10 seconds. The range is 1 to 25.</p> <p>Note To return to the default setting, use the no ip igmp query-max-response-time interface configuration command.</p>
Step 5	<p>end</p> <p>Example:</p> <pre>Device(config)# end</pre>	Returns to privileged EXEC mode.
Step 6	<p>show ip igmp interface [<i>interface-id</i>]</p> <p>Example:</p> <pre>Device# show ip igmp interface</pre>	Verifies your entries.
Step 7	<p>copy running-config startup-config</p> <p>Example:</p> <pre>Device# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Configuring the Device as a Statically Connected Member

At various times, either there is not a group member on a network segment or a host that cannot report its group membership by using IGMP. However, you may want multicast traffic to be sent to that network segment. The following commands are used to pull multicast traffic down to a network segment:

- **ip igmp join-group**—The device accepts the multicast packets in addition to forwarding them. Accepting the multicast packets prevents the device from fast switching.
- **ip igmp static-group**—The device does not accept the packets itself, but only forwards them. This method enables fast switching. The outgoing interface appears in the IGMP cache, but the device itself is not a member, as evidenced by lack of an L (local) flag in the multicast route entry.

This procedure is optional.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*

4. `ip igmp static-group group-address`
5. `end`
6. `show ip igmp interface [interface-id]`
7. `copy running-config startup-config`

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p>enable</p> <p>Example:</p> <pre>Device> enable</pre>	<p>Enables privileged EXEC mode.</p> <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	<p>configure terminal</p> <p>Example:</p> <pre>Device# configure terminal</pre>	<p>Enters global configuration mode.</p>
Step 3	<p>interface interface-id</p> <p>Example:</p> <pre>Device(config)# interface gigabitethernet 1/0/1</pre>	<p>Specifies the interface on which you want to enable multicast routing, and enters interface configuration mode.</p>
Step 4	<p>ip igmp static-group group-address</p> <p>Example:</p> <pre>Device(config-if)# ip igmp static-group 239.100.100.101</pre>	<p>Configures the device as a statically connected member of a group.</p> <p>By default, this feature is disabled.</p> <p>Note To remove the switch as a member of the group, use the no ip igmp static-group group-address interface configuration command.</p>
Step 5	<p>end</p> <p>Example:</p> <pre>Device(config)# end</pre>	<p>Returns to privileged EXEC mode.</p>
Step 6	<p>show ip igmp interface [interface-id]</p> <p>Example:</p> <pre>Device# show ip igmp interface gigabitethernet 1/0/1</pre>	<p>Verifies your entries.</p>

	Command or Action	Purpose
Step 7	copy running-config startup-config Example: Device# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Monitoring IGMP

You can display specific statistics, such as the contents of IP routing tables, caches, and databases.



Note This release does not support per-route statistics.

You can display information to learn resource usage and solve network problems. You can also display information about node reachability and discover the routing path that packets of your device are taking through the network.

You can use any of the privileged EXEC commands in the following table to display various routing statistics.

Table 19: Commands for Displaying System and Network Statistics

Command	Purpose
<code>show ip igmp groups [type-number detail]</code>	Displays the multicast groups that are directly connected to the device and that were learned through IGMP.
<code>show ip igmp interface [type number]</code>	Displays multicast-related information about an interface.
<code>show ip igmp profile [profile_number]</code>	Displays IGMP profile information.
<code>show ip igmp ssm-mapping [hostname/IP address]</code>	Displays IGMP SSM mapping information.
<code>show ip igmp static-group {class-map [interface [type]]</code>	Displays static group information.
<code>show ip igmp vrf</code>	Displays the selected VPN routing/forwarding instance by name.

Configuration Examples for IGMP

Example: Configuring the Device as a Member of a Multicast Group

This example shows how to enable the device to join multicast group 255.2.2.2:

```
Device(config)# interface gigabitethernet1/0/1
Device(config-if)# ip igmp join-group 255.2.2.2
Device(config-if)#
```

Example: Controlling Access to IP Multicast Groups

This example shows how to configure hosts attached to a port as able to join only group 255.2.2.2:

```
Switch(config)# access-list 1 255.2.2.2 0.0.0.0
Switch(config-if)# interface gigabitethernet1/0/1
Switch(config-if)# ip igmp access-group 1
```

Additional References

Related Documents

Related Topic	Document Title
For complete syntax and usage information for the commands used in this chapter.	<i>IP Multicast Command Reference, Cisco IOS Release 15.2(2)E (Catalyst 2960-XR Switch)</i>
Cisco IOS IP SLAs commands	Cisco IOS IP Multicast Command Reference

Error Message Decoder

Description	Link
To help you research and resolve system error messages in this release, use the Error Message Decoder tool.	https://www.cisco.com/cgi-bin/Support/Errordecoder/index.cgi

Standards and RFCs

Standard/RFC	Title
RFC 1112	Host Extensions for IP Multicasting
RFC 2236	Internet Group Management Protocol, Version 2
RFC 3376	<i>Internet Group Management Protocol, Version 3</i>

MIBs

MIB	MIBs Link
All the supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
<p>The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies.</p> <p>To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds.</p> <p>Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.</p>	http://www.cisco.com/support

Feature History and Information for IGMP

Release	Modification
Cisco IOS Release 15.0(2)EX1	This feature was introduced.



CHAPTER 14

Configuring IGMP Snooping and Multicast VLAN Registration

- [Prerequisites for Configuring IGMP Snooping and MVR, on page 137](#)
- [Restrictions for Configuring IGMP Snooping and MVR, on page 138](#)
- [Information About IGMP Snooping and MVR, on page 139](#)
- [How to Configure IGMP Snooping and MVR, on page 147](#)
- [Monitoring IGMP Snooping and MVR, on page 175](#)
- [Configuration Examples for IGMP Snooping and MVR, on page 178](#)
- [Additional References, on page 180](#)
- [Feature History and Information for IGMP Snooping, on page 181](#)

Prerequisites for Configuring IGMP Snooping and MVR

Prerequisites for IGMP Snooping

Observe these guidelines when configuring the IGMP snooping querier:

- Configure the VLAN in global configuration mode.
- Configure an IP address on the VLAN interface. When enabled, the IGMP snooping querier uses the IP address as the query source address.
- If there is no IP address configured on the VLAN interface, the IGMP snooping querier tries to use the configured global IP address for the IGMP querier. If there is no global IP address specified, the IGMP querier tries to use the VLAN device virtual interface (SVI) IP address (if one exists). If there is no SVI IP address, the device uses the first available IP address configured on the device. The first IP address available appears in the output of the **show ip interface** privileged EXEC command. The IGMP snooping querier does not generate an IGMP general query if it cannot find an available IP address on the device.
- The IGMP snooping querier supports IGMP Versions 1 and 2.
- When administratively enabled, the IGMP snooping querier moves to the nonquerier state if it detects the presence of a multicast router in the network.
- When it is administratively enabled, the IGMP snooping querier moves to the operationally disabled state under these conditions:

- IGMP snooping is disabled in the VLAN.
- PIM is enabled on the SVI of the corresponding VLAN.

•
•

Restrictions for Configuring IGMP Snooping and MVR

Restrictions for IGMP Snooping

The following are the restrictions for IGMP snooping:

- The switch supports homogeneous stacking, but does not support mixed stacking.
- The switch supports IGMPv3 snooping based only on the destination multicast MAC address. It does not support snooping based on the source MAC address or on proxy reports.
- IGMPv3 join and leave messages are not supported on devices running IGMP filtering or Multicast VLAN registration (MVR).
- IGMP report suppression is supported only when the multicast query has IGMPv1 and IGMPv2 reports. This feature is not supported when the query includes IGMPv3 reports.
- The IGMP configurable leave time is only supported on hosts running IGMP Version 2. IGMP version 2 is the default version for the device.

The actual leave latency in the network is usually the configured leave time. However, the leave time might vary around the configured time, depending on real-time CPU load conditions, network delays and the amount of traffic sent through the interface.

- The IGMP throttling action restriction can be applied only to Layer 2 ports. You can use **ip igmp max-groups action replace** interface configuration command on a logical EtherChannel interface but cannot use it on ports that belong to an EtherChannel port group.

When the maximum group limitation is set to the default (no maximum), entering the **ip igmp max-groups action {deny | replace}** command has no effect.

If you configure the throttling action and set the maximum group limitation after an interface has added multicast entries to the forwarding table, the forwarding-table entries are either aged out or removed, depending on the throttling action.

Restrictions for MVR

The following are restrictions for MVR:

- Only Layer 2 ports participate in MVR. You must configure ports as MVR receiver ports.
- Only one MVR multicast VLAN per device or device stack is supported.
- Receiver ports can only be access ports; they cannot be trunk ports. Receiver ports on a device can be in different VLANs, but should not belong to the multicast VLAN.

- The maximum number of multicast entries (MVR group addresses) that can be configured on a device (that is, the maximum number of television channels that can be received) is 256.
- MVR multicast data received in the source VLAN and leaving from receiver ports has its time-to-live (TTL) decremented by 1 in the device.
- Because MVR on the device uses IP multicast addresses instead of MAC multicast addresses, alias IP multicast addresses are allowed on the device. However, if the device is interoperating with Catalyst 3550 or Catalyst 3500 XL devices, you should not configure IP addresses that alias between themselves or with the reserved IP multicast addresses (in the range 224.0.0.xxx).
- Do not configure MVR on private VLAN ports.
- MVR is not supported when multicast routing is enabled on a device. If you enable multicast routing and a multicast routing protocol while MVR is enabled, MVR is disabled, and you receive a warning message. If you try to enable MVR while multicast routing and a multicast routing protocol are enabled, the operation to enable MVR is cancelled, and you receive an error message.
- MVR data received on an MVR receiver port is not forwarded to MVR source ports.
- MVR does not support IGMPv3 messages.

Information About IGMP Snooping and MVR

IGMP Snooping

Layer 2 devices can use IGMP snooping to constrain the flooding of multicast traffic by dynamically configuring Layer 2 interfaces so that multicast traffic is forwarded to only those interfaces associated with IP multicast devices. As the name implies, IGMP snooping requires the LAN device to snoop on the IGMP transmissions between the host and the router and to keep track of multicast groups and member ports. When the device receives an IGMP report from a host for a particular multicast group, the device adds the host port number to the forwarding table entry; when it receives an IGMP Leave Group message from a host, it removes the host port from the table entry. It also periodically deletes entries if it does not receive IGMP membership reports from the multicast clients.



Note For more information on IP multicast and IGMP, see RFC 1112 and RFC 2236.

The multicast router sends out periodic general queries to all VLANs. All hosts interested in this multicast traffic send join requests and are added to the forwarding table entry. The device creates one entry per VLAN in the IGMP snooping IP multicast forwarding table for each group from which it receives an IGMP join request.

The device supports IP multicast group-based bridging, instead of MAC-addressed based groups. With multicast MAC address-based groups, if an IP address being configured translates (aliases) to a previously configured MAC address or to any reserved multicast MAC addresses (in the range 224.0.0.xxx), the command fails. Because the device uses IP multicast groups, there are no address aliasing issues.

The IP multicast groups learned through IGMP snooping are dynamic. However, you can statically configure multicast groups by using the **ip igmp snooping vlan *vlan-id* static *ip_address* interface *interface-id*** global configuration command. If you specify group membership for a multicast group address statically, your setting

supersedes any automatic manipulation by IGMP snooping. Multicast group membership lists can consist of both user-defined and IGMP snooping-learned settings.

You can configure an IGMP snooping querier to support IGMP snooping in subnets without multicast interfaces because the multicast traffic does not need to be routed.

If a port spanning-tree, a port group, or a VLAN ID change occurs, the IGMP snooping-learned multicast groups from this port on the VLAN are deleted.

These sections describe IGMP snooping characteristics:

IGMP Versions

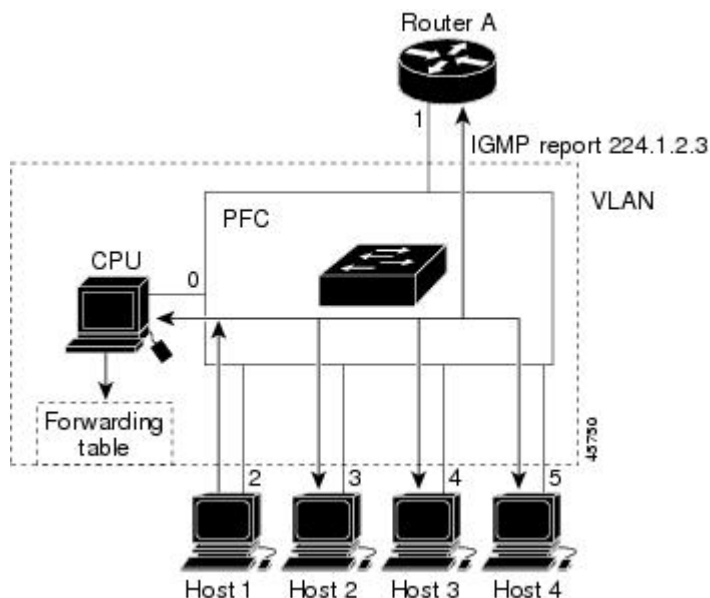
The device supports IGMP version 1, IGMP version 2, and IGMP version 3. These versions are interoperable on the device. For example, if IGMP snooping is enabled and the querier's version is IGMPv2, and the device receives an IGMPv3 report from a host, then the device can forward the IGMPv3 report to the multicast router.

An IGMPv3 device can receive messages from and forward messages to a device running the Source Specific Multicast (SSM) feature.

Joining a Multicast Group

Figure 7: Initial IGMP Join Message

When a host connected to the device wants to join an IP multicast group and it is an IGMP version 2 client, it sends an unsolicited IGMP join message, specifying the IP multicast group to join. Alternatively, when the device receives a general query from the router, it forwards the query to all ports in the VLAN. IGMP version 1 or version 2 hosts wanting to join the multicast group respond by sending a join message to the device. The device CPU creates a multicast forwarding-table entry for the group if it is not already present. The CPU also adds the interface where the join message was received to the forwarding-table entry. The host associated with that interface receives multicast traffic for that multicast group.



Router A sends a general query to the device, which forwards the query to ports 2 through 5, all of which are members of the same VLAN. Host 1 wants to join multicast group 224.1.2.3 and multicasts an IGMP

membership report (IGMP join message) to the group. The device CPU uses the information in the IGMP report to set up a forwarding-table entry that includes the port numbers connected to Host 1 and to the router.

Table 20: IGMP Snooping Forwarding Table

Destination Address	Type of Packet	Ports
224.1.2.3	IGMP	1, 2

The device hardware can distinguish IGMP information packets from other packets for the multicast group. The information in the table tells the switching engine to send frames addressed to the 224.1.2.3 multicast IP address that are not IGMP packets to the router and to the host that has joined the group.

Figure 8: Second Host Joining a Multicast Group

If another host (for example, Host 4) sends an unsolicited IGMP join message for the same group, the CPU receives that message and adds the port number of Host 4 to the forwarding table. Because the forwarding table directs IGMP messages only to the CPU, the message is not flooded to other ports on the device. Any known multicast traffic is forwarded to the group and not to the CPU.

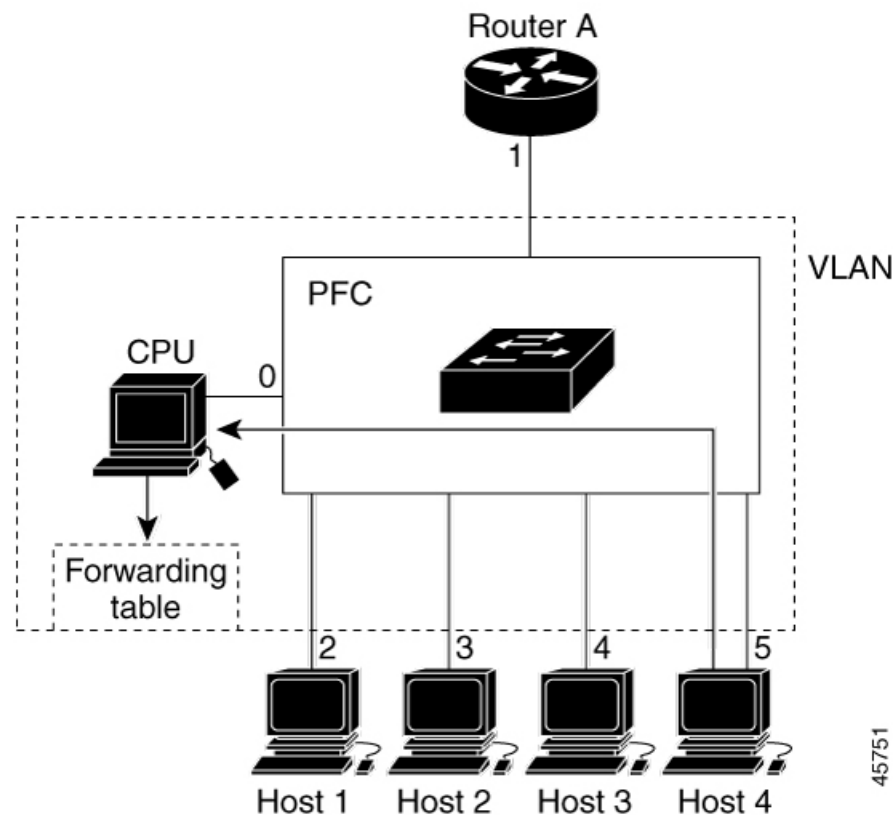


Table 21: Updated IGMP Snooping Forwarding Table

Destination Address	Type of Packet	Ports
224.1.2.3	IGMP	1, 2, 5

Leaving a Multicast Group

The router sends periodic multicast general queries, and the device forwards these queries through all ports in the VLAN. Interested hosts respond to the queries. If at least one host in the VLAN wants to receive multicast traffic, the router continues forwarding the multicast traffic to the VLAN. The device forwards multicast group traffic only to those hosts listed in the forwarding table for that IP multicast group maintained by IGMP snooping.

When hosts want to leave a multicast group, they can silently leave, or they can send a leave message. When the device receives a leave message from a host, it sends a group-specific query to learn if any other devices connected to that interface are interested in traffic for the specific multicast group. The device then updates the forwarding table for that MAC group so that only those hosts interested in receiving multicast traffic for the group are listed in the forwarding table. If the router receives no reports from a VLAN, it removes the group for the VLAN from its IGMP cache.

Immediate Leave

The device uses IGMP snooping Immediate Leave to remove from the forwarding table an interface that sends a leave message without the device sending group-specific queries to the interface. The VLAN interface is pruned from the multicast tree for the multicast group specified in the original leave message. Immediate Leave ensures optimal bandwidth management for all hosts on a switched network, even when multiple multicast groups are simultaneously in use.

Immediate Leave is only supported on IGMP version 2 hosts. IGMP version 2 is the default version for the device.



Note You should use the Immediate Leave feature only on VLANs where a single host is connected to each port. If Immediate Leave is enabled on VLANs where more than one host is connected to a port, some hosts may be dropped inadvertently.

IGMP Configurable-Leave Timer

You can configure the time that the device waits after sending a group-specific query to determine if hosts are still interested in a specific multicast group. The IGMP leave response time can be configured from 100 to 32767 milliseconds.

IGMP Report Suppression



Note IGMP report suppression is supported only when the multicast query has IGMPv1 and IGMPv2 reports. This feature is not supported when the query includes IGMPv3 reports.

The device uses IGMP report suppression to forward only one IGMP report per multicast router query to multicast devices. When IGMP report suppression is enabled (the default), the device sends the first IGMP report from all hosts for a group to all the multicast routers. The device does not send the remaining IGMP reports for the group to the multicast routers. This feature prevents duplicate reports from being sent to the multicast devices.

If the multicast router query includes requests only for IGMPv1 and IGMPv2 reports, the device forwards only the first IGMPv1 or IGMPv2 report from all hosts for a group to all the multicast routers.

If the multicast router query also includes requests for IGMPv3 reports, the device forwards all IGMPv1, IGMPv2, and IGMPv3 reports for a group to the multicast devices.

If you disable IGMP report suppression, all IGMP reports are forwarded to the multicast routers.

IGMP Snooping and Device Stacks

IGMP snooping functions across the device stack; that is, IGMP control information from one device is distributed to all devices in the stack. Regardless of the stack member through which IGMP multicast data enters the stack, the data reaches the hosts that have registered for that group.

If a device in the stack fails or is removed from the stack, only the members of the multicast group that are on that device will not receive the multicast data. All other members of a multicast group on other devices in the stack continue to receive multicast data streams. However, multicast groups that are common for both Layer 2 and Layer 3 (IP multicast routing) might take longer to converge if the active device is removed.

Default IGMP Snooping Configuration

This table displays the default IGMP snooping configuration for the device.

Table 22: Default IGMP Snooping Configuration

Feature	Default Setting
IGMP snooping	Enabled globally and per VLAN
Multicast routers	None configured
IGMP snooping Immediate Leave	Disabled
Static groups	None configured
TCN ¹ flood query count	2
TCN query solicitation	Disabled
IGMP snooping querier	Disabled
IGMP report suppression	Enabled

¹ (1) TCN = Topology Change Notification

Multicast VLAN Registration

Multicast VLAN Registration (MVR) is designed for applications using wide-scale deployment of multicast traffic across an Ethernet ring-based service-provider network (for example, the broadcast of multiple television channels over a service-provider network). MVR allows a subscriber on a port to subscribe and unsubscribe to a multicast stream on the network-wide multicast VLAN. It allows the single multicast VLAN to be shared in the network while subscribers remain in separate VLANs. MVR provides the ability to continuously send multicast streams in the multicast VLAN, but to isolate the streams from the subscriber VLANs for bandwidth and security reasons.

These sections describe MVR:

MVR and IGMP



Note MVR can coexist with IGMP snooping on a device.

MVR assumes that subscriber ports subscribe and unsubscribe (join and leave) these multicast streams by sending out IGMP join and leave messages. These messages can originate from an IGMP version-2-compatible host with an Ethernet connection. Although MVR operates on the underlying method of IGMP snooping, the two features operate independently of each other. One can be enabled or disabled without affecting the behavior of the other feature. However, if IGMP snooping and MVR are both enabled, MVR reacts only to join and leave messages from multicast groups configured under MVR. Join and leave messages from all other multicast groups are managed by IGMP snooping.

The device CPU identifies the MVR IP multicast streams and their associated IP multicast group in the device forwarding table, intercepts the IGMP messages, and modifies the forwarding table to include or remove the subscriber as a receiver of the multicast stream, even though the receivers might be in a different VLAN from the source. This forwarding behavior selectively allows traffic to cross between different VLANs.

Modes of Operation

You can set the device for compatible or dynamic mode of MVR operation:

- In compatible mode, multicast data received by MVR hosts is forwarded to all MVR data ports, regardless of MVR host membership on those ports. The multicast data is forwarded only to those receiver ports that MVR hosts have joined, either by IGMP reports or by MVR static configuration. IGMP reports received from MVR hosts are never forwarded from MVR data ports that were configured in the device.
- In dynamic mode, multicast data received by MVR hosts on the device is forwarded from only those MVR data and client ports that the MVR hosts have joined, either by IGMP reports or by MVR static configuration. Any IGMP reports received from MVR hosts are also forwarded from all the MVR data ports in the host. This eliminates using unnecessary bandwidth on MVR data port links, which occurs when the device runs in compatible mode.

MVR and Switch Stacks

Only one MVR multicast VLAN per device or device stack is supported.

Receiver ports and source ports can be on different devices in a device stack. Multicast data sent on the multicast VLAN is forwarded to all MVR receiver ports across the stack. When a new device is added to a stack, by default it has no receiver ports.

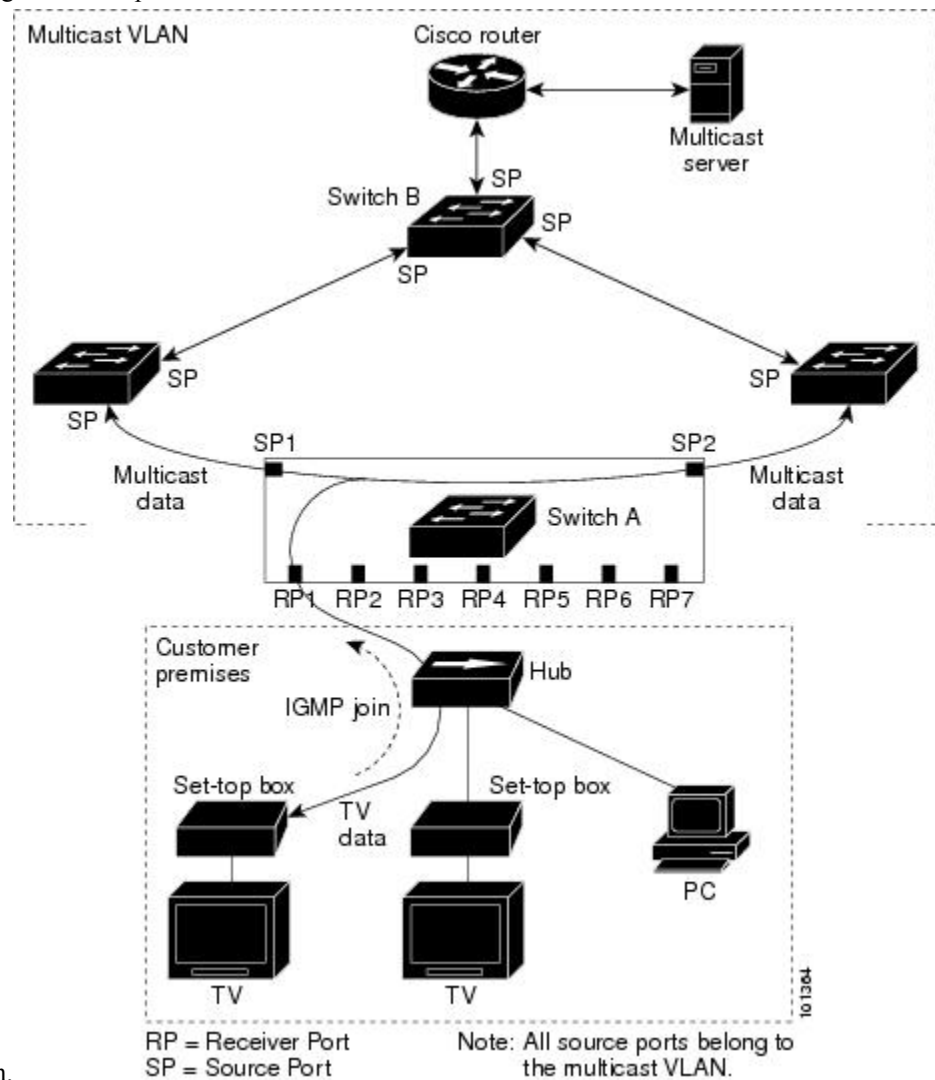
If a device fails or is removed from the stack, only those receiver ports belonging to that device will not receive the multicast data. All other receiver ports on other devices continue to receive the multicast data.

MVR in a Multicast Television Application

In a multicast television application, a PC or a television with a set-top box can receive the multicast stream. Multiple set-top boxes or PCs can be connected to one subscriber port, which is a device port configured as an MVR receiver port.

Figure 9: Multicast VLAN Registration Example

The following is an example



configuration.

In this example configuration, DHCP assigns an IP address to the set-top box or the PC. When a subscriber selects a channel, the set-top box or PC sends an IGMP report to Switch A to join the appropriate multicast. If the IGMP report matches one of the configured IP multicast group addresses, the device CPU modifies the hardware address table to include this receiver port and VLAN as a forwarding destination of the specified multicast stream when it is received from the multicast VLAN. Uplink ports that send and receive multicast data to and from the multicast VLAN are called MVR source ports.

When a subscriber changes channels or turns off the television, the set-top box sends an IGMP leave message for the multicast stream. The device CPU sends a MAC-based general query through the receiver port VLAN. If there is another set-top box in the VLAN still subscribing to this group, that set-top box must respond within the maximum response time specified in the query. If the CPU does not receive a response, it eliminates the receiver port as a forwarding destination for this group.

Without Immediate Leave, when the device receives an IGMP leave message from a subscriber on a receiver port, it sends out an IGMP query on that port and waits for IGMP group membership reports. If no reports

are received in a configured time period, the receiver port is removed from multicast group membership. With Immediate Leave, an IGMP query is not sent from the receiver port on which the IGMP leave was received. As soon as the leave message is received, the receiver port is removed from multicast group membership, which speeds up leave latency. Enable the Immediate-Leave feature only on receiver ports to which a single receiver device is connected.

MVR eliminates the need to duplicate television-channel multicast traffic for subscribers in each VLAN. Multicast traffic for all channels is only sent around the VLAN trunk once—only on the multicast VLAN. The IGMP leave and join messages are in the VLAN to which the subscriber port is assigned. These messages dynamically register for streams of multicast traffic in the multicast VLAN on the Layer 3 device. The access layer device, Switch A, modifies the forwarding behavior to allow the traffic to be forwarded from the multicast VLAN to the subscriber port in a different VLAN, selectively allowing traffic to cross between two VLANs.

IGMP reports are sent to the same IP multicast group address as the multicast data. The Switch A CPU must capture all IGMP join and leave messages from receiver ports and forward them to the multicast VLAN of the source (uplink) port, based on the MVR mode.

Default MVR Configuration

Table 23: Default MVR Configuration

Feature	Default Setting
MVR	Disabled globally and per interface
Multicast addresses	None configured
Query response time	0.5 second
Multicast VLAN	VLAN 1
Mode	Compatible
Interface (per port) default	Neither a receiver nor a source port
Immediate Leave	Disabled on all ports

IGMP Filtering and Throttling

In some environments, for example, metropolitan or multiple-dwelling unit (MDU) installations, you might want to control the set of multicast groups to which a user on a device port can belong. You can control the distribution of multicast services, such as IP/TV, based on some type of subscription or service plan. You might also want to limit the number of multicast groups to which a user on a device port can belong.

With the IGMP filtering feature, you can filter multicast joins on a per-port basis by configuring IP multicast profiles and associating them with individual device ports. An IGMP profile can contain one or more multicast groups and specifies whether access to the group is permitted or denied. If an IGMP profile denying access to a multicast group is applied to a device port, the IGMP join report requesting the stream of IP multicast traffic is dropped, and the port is not allowed to receive IP multicast traffic from that group. If the filtering action permits access to the multicast group, the IGMP report from the port is forwarded for normal processing. You can also set the maximum number of IGMP groups that a Layer 2 interface can join.

IGMP filtering controls only group-specific query and membership reports, including join and leave reports. It does not control general IGMP queries. IGMP filtering has no relationship with the function that directs

the forwarding of IP multicast traffic. The filtering feature operates in the same manner whether CGMP or MVR is used to forward the multicast traffic.

IGMP filtering applies only to the dynamic learning of IP multicast group addresses, not static configuration.

With the IGMP throttling feature, you can set the maximum number of IGMP groups that a Layer 2 interface can join. If the maximum number of IGMP groups is set, the IGMP snooping forwarding table contains the maximum number of entries, and the interface receives an IGMP join report, you can configure an interface to drop the IGMP report or to replace the randomly selected multicast entry with the received IGMP report.



Note IGMPv3 join and leave messages are not supported on devices running IGMP filtering.

Default IGMP Filtering and Throttling Configuration

This table displays the default IGMP filtering and throttling configuration for the device.

Table 24: Default IGMP Filtering Configuration

Feature	Default Setting
IGMP filters	None applied.
IGMP maximum number of IGMP groups	No maximum set. Note When the maximum number of groups is in the forwarding table, the default IGMP throttling action is to deny the IGMP report.
IGMP profiles	None defined.
IGMP profile action	Deny the range addresses.

How to Configure IGMP Snooping and MVR

Enabling or Disabling IGMP Snooping on a Device

When IGMP snooping is globally enabled or disabled, it is also enabled or disabled in all existing VLAN interfaces. IGMP snooping is enabled on all VLANs by default, but can be enabled and disabled on a per-VLAN basis.

Global IGMP snooping overrides the VLAN IGMP snooping. If global snooping is disabled, you cannot enable VLAN snooping. If global snooping is enabled, you can enable or disable VLAN snooping.

Follow these steps to globally enable IGMP snooping on the device:

SUMMARY STEPS

1. enable

2. **configure terminal**
3. **ip igmp snooping**
4. **end**
5. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	ip igmp snooping Example: Device(config)# ip igmp snooping	Globally enables IGMP snooping in all existing VLAN interfaces. Note To globally disable IGMP snooping on all VLAN interfaces, use the no ip igmp snooping global configuration command.
Step 4	end Example: Device(config)# end	Returns to privileged EXEC mode.
Step 5	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Enabling or Disabling IGMP Snooping on a VLAN Interface

Follow these steps to enable IGMP snooping on a VLAN interface:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip igmp snooping vlan *vlan-id***
4. **end**

5. copy running-config startup-config

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	ip igmp snooping vlan <i>vlan-id</i> Example: Device(config)# ip igmp snooping vlan 7	Enables IGMP snooping on the VLAN interface. The VLAN ID range is 1 to 1001 and 1006 to 4094. IGMP snooping must be globally enabled before you can enable VLAN snooping. Note To disable IGMP snooping on a VLAN interface, use the no ip igmp snooping vlan <i>vlan-id</i> global configuration command for the specified VLAN number.
Step 4	end Example: Device(config)# end	Returns to privileged EXEC mode.
Step 5	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Setting the Snooping Method

Multicast-capable router ports are added to the forwarding table for every Layer 2 multicast entry. The switch learns of the ports through one of these methods:

- Snooping on IGMP queries, Protocol-Independent Multicast (PIM) packets, and Distance Vector Multicast Routing Protocol (DVMRP) packets.
- Listening to Cisco Group Management Protocol (CGMP) packets from other routers.
- Statically connecting to a multicast router port using the **ip igmp snooping mrouter** global configuration command.

You can configure the switch either to snoop on IGMP queries and PIM/DVMRP packets or to listen to CGMP self-join or proxy-join packets. By default, the switch snoops on PIM/DVMRP packets on all VLANs. To learn of multicast router ports through only CGMP packets, use the **ip igmp snooping vlan *vlan-id* mrouter learn cgmp** global configuration command. When this command is entered, the router listens to only CGMP self-join and CGMP proxy-join packets and to no other CGMP packets. To learn of multicast router ports through only PIM-DVMRP packets, use the **ip igmp snooping vlan *vlan-id* mrouter learn pim-dvmrp** global configuration command.

If you want to use CGMP as the learning method and no multicast routers in the VLAN are CGMP proxy-enabled, you must enter the **ip cgmp router-only** command to dynamically access the router.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip igmp snooping vlan *vlan-id* mrouter learn {cgmp | pim-dvmrp }**
4. **end**
5. **show ip igmp snooping**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Device> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: <pre>Device# configure terminal</pre>	Enters global configuration mode.
Step 3	ip igmp snooping vlan <i>vlan-id</i> mrouter learn {cgmp pim-dvmrp } Example: <pre>Device(config)# ip igmp snooping vlan 1 mrouter learn cgmp</pre>	Specifies the multicast router learning method: <ul style="list-style-type: none"> • cgmp—Listens for CGMP packets. This method is useful for reducing control traffic. • pim-dvmrp—Snoops on IGMP queries and PIM-DVMRP packets. This is the default. <p>Note To return to the default learning method, use the no ip igmp snooping vlan <i>vlan-id</i> mrouter learn cgmp global configuration command.</p>
Step 4	end Example:	Returns to privileged EXEC mode.

	Command or Action	Purpose
	Device(config)# end	
Step 5	show ip igmp snooping Example: Device# show ip igmp snooping	Verifies the configuration.
Step 6	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Configuring a Multicast Router Port

Perform these steps to add a multicast router port (enable a static connection to a multicast router) on the device.



Note Static connections to multicast routers are supported only on device ports.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip igmp snooping vlan *vlan-id* mrouter interface *interface-id***
4. **end**
5. **show ip igmp snooping mrouter [vlan *vlan-id*]**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example:	Enters global configuration mode.

	Command or Action	Purpose
	Device# <code>configure terminal</code>	
Step 3	<p>ip igmp snooping vlan <i>vlan-id</i> mrouter interface <i>interface-id</i></p> <p>Example:</p> <pre>Device(config)# ip igmp snooping vlan 5 mrouter interface gigabitethernet1/0/1</pre>	<p>Specifies the multicast router VLAN ID and the interface to the multicast router.</p> <ul style="list-style-type: none"> • The VLAN ID range is 1 to 1001 and 1006 to 4094. • The interface can be a physical interface or a port channel. The port-channel range is 1 to 128. <p>Note To remove a multicast router port from the VLAN, use the no ip igmp snooping vlan <i>vlan-id</i> mrouter interface <i>interface-id</i> global configuration command.</p>
Step 4	<p>end</p> <p>Example:</p> <pre>Device(config)# end</pre>	Returns to privileged EXEC mode.
Step 5	<p>show ip igmp snooping mrouter [vlan <i>vlan-id</i>]</p> <p>Example:</p> <pre>Device# show ip igmp snooping mrouter vlan 5</pre>	Verifies that IGMP snooping is enabled on the VLAN interface.
Step 6	<p>copy running-config startup-config</p> <p>Example:</p> <pre>Device# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Configuring a Host Statically to Join a Group

Hosts or Layer 2 ports normally join multicast groups dynamically, but you can also statically configure a host on an interface.

Follow these steps to add a Layer 2 port as a member of a multicast group:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip igmp snooping vlan *vlan-id* static *ip_address* interface *interface-id***
4. **end**
5. **show ip igmp snooping groups**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	ip igmp snooping vlan <i>vlan-id</i> static <i>ip_address</i> interface <i>interface-id</i> Example: Device(config)# ip igmp snooping vlan 105 static 230.0.0.1 interface gigabitethernet1/0/1	Statically configures a Layer 2 port as a member of a multicast group: <ul style="list-style-type: none"> • <i>vlan-id</i> is the multicast group VLAN ID. The range is 1 to 1001 and 1006 to 4094. • <i>ip-address</i> is the group IP address. • <i>interface-id</i> is the member port. It can be a physical interface or a port channel (1 to 128). <p>Note To remove the Layer 2 port from the multicast group, use the no ip igmp snooping vlan <i>vlan-id</i> static <i>mac-address</i> interface <i>interface-id</i> global configuration command.</p>
Step 4	end Example: Device(config)# end	Returns to privileged EXEC mode.
Step 5	show ip igmp snooping groups Example: Device# show ip igmp snooping groups	Verifies the member port and the IP address.
Step 6	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Enabling IGMP Immediate Leave

When you enable IGMP Immediate Leave, the device immediately removes a port when it detects an IGMP Version 2 leave message on that port. You should use the Immediate-Leave feature only when there is a single receiver present on every port in the VLAN.



Note Immediate Leave is supported only on IGMP Version 2 hosts. IGMP Version 2 is the default version for the device.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip igmp snooping vlan *vlan-id* immediate-leave**
4. **end**
5. **show ip igmp snooping vlan *vlan-id***
6. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	ip igmp snooping vlan <i>vlan-id</i> immediate-leave Example: Device(config)# ip igmp snooping vlan 21 immediate-leave	Enables IGMP Immediate Leave on the VLAN interface. <p>Note To disable IGMP Immediate Leave on a VLAN, use the no ip igmp snooping vlan <i>vlan-id</i> immediate-leave global configuration command.</p>
Step 4	end Example: Device(config)# end	Returns to privileged EXEC mode.

	Command or Action	Purpose
Step 5	show ip igmp snooping vlan <i>vlan-id</i> Example: <pre>Device# show ip igmp snooping vlan 21</pre>	Verifies that Immediate Leave is enabled on the VLAN interface.
Step 6	end Example: <pre>Device(config)# end</pre>	Returns to privileged EXEC mode.

Configuring the IGMP Leave Timer

You can configure the leave time globally or on a per-VLAN basis. Follow these steps to enable the IGMP configurable-leave timer:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip igmp snooping last-member-query-interval *time***
4. **ip igmp snooping vlan *vlan-id* last-member-query-interval *time***
5. **end**
6. **show ip igmp snooping**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Device> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: <pre>Device# configure terminal</pre>	Enters global configuration mode.
Step 3	ip igmp snooping last-member-query-interval <i>time</i> Example: <pre>Device(config)# ip igmp snooping</pre>	Configures the IGMP leave timer globally. The range is 100 to 32767 milliseconds. The default leave time is 1000 milliseconds.

	Command or Action	Purpose
	<code>last-member-query-interval 1000</code>	Note To globally reset the IGMP leave timer to the default setting, use the no ip igmp snooping last-member-query-interval global configuration command.
Step 4	<p>ip igmp snooping vlan <i>vlan-id</i> last-member-query-interval <i>time</i></p> <p>Example:</p> <pre>Device(config)# ip igmp snooping vlan 210 last-member-query-interval 1000</pre>	<p>(Optional) Configures the IGMP leave time on the VLAN interface. The range is 100 to 32767 milliseconds.</p> <p>Note Configuring the leave time on a VLAN overrides the globally configured timer.</p> <p>Note To remove the configured IGMP leave-time setting from the specified VLAN, use the no ip igmp snooping vlan <i>vlan-id</i> last-member-query-interval global configuration command.</p>
Step 5	<p>end</p> <p>Example:</p> <pre>Device(config)# end</pre>	Returns to privileged EXEC mode.
Step 6	<p>show ip igmp snooping</p> <p>Example:</p> <pre>Device# show ip igmp snooping</pre>	(Optional) Displays the configured IGMP leave time.
Step 7	<p>copy running-config startup-config</p> <p>Example:</p> <pre>Device# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Configuring TCN-Related Commands

Controlling the Multicast Flooding Time After a TCN Event

You can configure the number of general queries by which multicast data traffic is flooded after a topology change notification (TCN) event. If you set the TCN flood query count to 1 the flooding stops after receiving 1 general query. If you set the count to 7, the flooding continues until 7 general queries are received. Groups are relearned based on the general queries received during the TCN event.

Some examples of TCN events are when the client location is changed and the receiver is on same port that was blocked but is now forwarding, and when a port goes down without sending a leave message.

Follow these steps to configure the TCN flood query count:

SUMMARY STEPS

1. enable
2. configure terminal
3. ip igmp snooping tcn flood query count *count*
4. end
5. show ip igmp snooping
6. copy running-config startup-config

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	ip igmp snooping tcn flood query count <i>count</i> Example: Device(config)# ip igmp snooping tcn flood query count 3	Specifies the number of IGMP general queries for which the multicast traffic is flooded. The range is 1 to 10. The default, the flooding query count is 2. Note To return to the default flooding query count, use the no ip igmp snooping tcn flood query count global configuration command.
Step 4	end Example: Device(config)# end	Returns to privileged EXEC mode.
Step 5	show ip igmp snooping Example: Device# show ip igmp snooping	Verifies the TCN settings.
Step 6	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Recovering from Flood Mode

When a topology change occurs, the spanning-tree root sends a special IGMP leave message (also known as global leave) with the group multicast address 0.0.0.0. However, you can enable the device to send the global leave message whether it is the spanning-tree root or not. When the router receives this special leave, it immediately sends general queries, which expedite the process of recovering from the flood mode during the TCN event. Leaves are always sent if the device is the spanning-tree root regardless of this configuration.

Follow these steps to enable sending of leave messages:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip igmp snooping tcn query solicit**
4. **end**
5. **show ip igmp snooping**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	ip igmp snooping tcn query solicit Example: Device(config)# ip igmp snooping tcn query solicit	Sends an IGMP leave message (global leave) to speed the process of recovering from the flood mode caused during a TCN event. By default, query solicitation is disabled. Note To return to the default query solicitation, use the no ip igmp snooping tcn query solicit global configuration command.
Step 4	end Example: Device(config)# end	Returns to privileged EXEC mode.
Step 5	show ip igmp snooping Example:	Verifies the TCN settings.

	Command or Action	Purpose
	Device# <code>show ip igmp snooping</code>	
Step 6	copy running-config startup-config Example: Device# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Disabling Multicast Flooding During a TCN Event

When the device receives a TCN, multicast traffic is flooded to all the ports until 2 general queries are received. If the device has many ports with attached hosts that are subscribed to different multicast groups, this flooding might exceed the capacity of the link and cause packet loss. Follow these steps to control TCN flooding:

SUMMARY STEPS

1. `enable`
2. `configure terminal`
3. `interface interface-id`
4. `no ip igmp snooping tcn flood`
5. `end`
6. `show ip igmp snooping`
7. `copy running-config startup-config`

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> <code>enable</code>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode.
Step 3	interface interface-id Example: Device(config)# <code>interface gigabitethernet 1/0/1</code>	Specifies the interface to be configured, and enters interface configuration mode.

	Command or Action	Purpose
Step 4	no ip igmp snooping tcn flood Example: <pre>Device(config-if)# no ip igmp snooping tcn flood</pre>	Disables the flooding of multicast traffic during a spanning-tree TCN event. By default, multicast flooding is enabled on an interface. Note To re-enable multicast flooding on an interface, use the ip igmp snooping tcn flood interface configuration command.
Step 5	end Example: <pre>Device(config)# end</pre>	Returns to privileged EXEC mode.
Step 6	show ip igmp snooping Example: <pre>Device# show ip igmp snooping</pre>	Verifies the TCN settings.
Step 7	copy running-config startup-config Example: <pre>Device# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Configuring the IGMP Snooping Querier

Follow these steps to enable the IGMP snooping querier feature in a VLAN:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip igmp snooping querier**
4. **ip igmp snooping querier address *ip_address***
5. **ip igmp snooping querier query-interval *interval-count***
6. **ip igmp snooping querier tcn query [count *count* | interval *interval*]**
7. **ip igmp snooping querier timer expiry *timeout***
8. **ip igmp snooping querier version *version***
9. **end**
10. **show ip igmp snooping vlan *vlan-id***
11. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	ip igmp snooping querier Example: Device(config)# ip igmp snooping querier	Enables the IGMP snooping querier.
Step 4	ip igmp snooping querier address <i>ip_address</i> Example: Device(config)# ip igmp snooping querier address 172.16.24.1	(Optional) Specifies an IP address for the IGMP snooping querier. If you do not specify an IP address, the querier tries to use the global IP address configured for the IGMP querier. Note The IGMP snooping querier does not generate an IGMP general query if it cannot find an IP address on the device.
Step 5	ip igmp snooping querier query-interval <i>interval-count</i> Example: Device(config)# ip igmp snooping querier query-interval 30	(Optional) Sets the interval between IGMP queriers. The range is 1 to 18000 seconds.
Step 6	ip igmp snooping querier tcn query [count <i>count</i> interval <i>interval</i>] Example: Device(config)# ip igmp snooping querier tcn query interval 20	(Optional) Sets the time between Topology Change Notification (TCN) queries. The count range is 1 to 10. The interval range is 1 to 255 seconds.
Step 7	ip igmp snooping querier timer expiry <i>timeout</i> Example: Device(config)# ip igmp snooping querier timer expiry 180	(Optional) Sets the length of time until the IGMP querier expires. The range is 60 to 300 seconds.

	Command or Action	Purpose
Step 8	ip igmp snooping querier version <i>version</i> Example: <pre>Device(config)# ip igmp snooping querier version 2</pre>	(Optional) Selects the IGMP version number that the querier feature uses. Select 1 or 2.
Step 9	end Example: <pre>Device(config)# end</pre>	Returns to privileged EXEC mode.
Step 10	show ip igmp snooping vlan <i>vlan-id</i> Example: <pre>Device# show ip igmp snooping vlan 30</pre>	(Optional) Verifies that the IGMP snooping querier is enabled on the VLAN interface. The VLAN ID range is 1 to 1001 and 1006 to 4094.
Step 11	copy running-config startup-config Example: <pre>Device# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Disabling IGMP Report Suppression

Follow these steps to disable IGMP report suppression:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **no ip igmp snooping report-suppression**
4. **end**
5. **show ip igmp snooping**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Device> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.

	Command or Action	Purpose
Step 2	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode.
Step 3	no ip igmp snooping report-suppression Example: Device(config)# <code>no ip igmp snooping report-suppression</code>	Disables IGMP report suppression. When report suppression is disabled, all IGMP reports are forwarded to the multicast routers. IGMP report suppression is enabled by default. When IGMP report suppression is enabled, the device forwards only one IGMP report per multicast router query. Note To re-enable IGMP report suppression, use the ip igmp snooping report-suppression global configuration command.
Step 4	end Example: Device(config)# <code>end</code>	Returns to privileged EXEC mode.
Step 5	show ip igmp snooping Example: Device# <code>show ip igmp snooping</code>	Verifies that IGMP report suppression is disabled.
Step 6	copy running-config startup-config Example: Device# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Configuring MVR Global Parameters

You do not need to set the optional MVR parameters if you choose to use the default settings. If you want to change the default parameters (except for the MVR VLAN), you must first enable MVR.



Note For complete syntax and usage information for the commands used in this section, see the command reference for this release.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **mvr**
4. **mvr group** *ip-address* [*count*]
5. **mvr querytime** *value*
6. **mvr vlan** *vlan-id*
7. **mvr mode** {**dynamic** | **compatible**}
8. **end**
9. Use one of the following:
 - **show mvr**
 - **show mvr members**
10. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	mvr Example: Device (config)# mvr	Enables MVR on the device.
Step 4	mvr group <i>ip-address</i> [<i>count</i>] Example: Device(config)# mvr group 228.1.23.4	Configures an IP multicast address on the device or use the <i>count</i> parameter to configure a contiguous series of MVR group addresses (the range for <i>count</i> is 1 to 256; the default is 1). Any multicast data sent to this address is sent to all source ports on the device and all receiver ports that have elected to receive data on that multicast address. Each multicast address would correspond to one television channel. Note To return the switch to its default settings, use the no mvr [mode group <i>ip-address</i> querytime vlan] global configuration commands.

	Command or Action	Purpose
Step 5	mvr querytime <i>value</i> Example: <pre>Device(config)# mvr querytime 10</pre>	(Optional) Defines the maximum time to wait for IGMP report memberships on a receiver port before removing the port from multicast group membership. The value is in units of tenths of a second. The range is 1 to 100, and the default is 5 tenths or one-half second.
Step 6	mvr vlan <i>vlan-id</i> Example: <pre>Device(config)# mvr vlan 22</pre>	(Optional) Specifies the VLAN in which multicast data is received; all source ports must belong to this VLAN. The VLAN range is 1 to 1001 and 1006 to 4094. The default is VLAN 1.
Step 7	mvr mode { dynamic compatible } Example: <pre>Device(config)# mvr mode dynamic</pre>	(Optional) Specifies the MVR mode of operation: <ul style="list-style-type: none"> • dynamic—Allows dynamic MVR membership on source ports. • compatible—Is compatible with Catalyst 3500 XL and Catalyst 2900 XL devices and does not support IGMP dynamic joins on source ports. The default is compatible mode. Note To return the switch to its default settings, use the no mvr [mode group ip-address querytime vlan] global configuration commands.
Step 8	end Example: <pre>Device(config)# end</pre>	Returns to privileged EXEC mode.
Step 9	Use one of the following: <ul style="list-style-type: none"> • show mvr • show mvr members Example: <pre>Device# show mvr</pre> OR <pre>Device# show mvr members</pre>	Verifies the configuration.
Step 10	copy running-config startup-config Example:	(Optional) Saves your entries in the configuration file.

	Command or Action	Purpose
	Device# <code>copy running-config startup-config</code>	

Configuring MVR Interfaces

Follow these steps to configure Layer 2 MVR interfaces:

SUMMARY STEPS

1. `enable`
2. `configure terminal`
3. `mvr`
4. `interface interface-id`
5. `mvr type {source | receiver}`
6. `mvr vlan vlan-id group [ip-address]`
7. `mvr immediate`
8. `end`
9. Use one of the following:
 - `show mvr`
 - `show mvr interface`
 - `show mvr members`
10. `copy running-config startup-config`

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> <code>enable</code>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode.
Step 3	mvr Example: Device (config)# <code>mvr</code>	Enables MVR on the device.

	Command or Action	Purpose
Step 4	interface <i>interface-id</i> Example: <pre>Device(config)# interface gigabitethernet1/0/2</pre>	Specifies the Layer 2 port to configure, and enter interface configuration mode.
Step 5	mvr type {source receiver} Example: <pre>Device(config-if)# mvr type receiver</pre>	<p>Configures an MVR port as one of these:</p> <ul style="list-style-type: none"> • source—Configures uplink ports that receive and send multicast data as source ports. Subscribers cannot be directly connected to source ports. All source ports on a device belong to the single multicast VLAN. • receiver—Configures a port as a receiver port if it is a subscriber port and should only receive multicast data. It does not receive data unless it becomes a member of the multicast group, either statically or by using IGMP leave and join messages. Receiver ports cannot belong to the multicast VLAN. <p>The default configuration is as a non-MVR port. If you attempt to configure a non-MVR port with MVR characteristics, the operation fails.</p> <p>Note To return the interface to its default settings, use the no mvr [type immediate vlan <i>vlan-id</i> group] interface configuration commands.</p>
Step 6	mvr vlan <i>vlan-id</i> group [<i>ip-address</i>] Example: <pre>Device(config-if)# mvr vlan 22 group 228.1.23.4</pre>	<p>(Optional) Statically configures a port to receive multicast traffic sent to the multicast VLAN and the IP multicast address. A port statically configured as a member of a group remains a member of the group until statically removed.</p> <p>Note In compatible mode, this command applies to only receiver ports. In dynamic mode, it applies to receiver ports and source ports.</p> <p>Receiver ports can also dynamically join multicast groups by using IGMP join and leave messages.</p>
Step 7	mvr immediate Example: <pre>Device(config-if)# mvr immediate</pre>	<p>(Optional) Enables the Immediate-Leave feature of MVR on the port.</p> <p>Note This command applies to only receiver ports and should only be enabled on receiver ports to which a single receiver device is connected.</p>

	Command or Action	Purpose
Step 8	end Example: Device(config)# end	Returns to privileged EXEC mode.
Step 9	Use one of the following: <ul style="list-style-type: none"> • show mvr • show mvr interface • show mvr members Example: <pre> Device# show mvr interface Port Type Status Immediate Leave ----- ----- Gi1/0/2 RECEIVER ACTIVE/DOWN ENABLED </pre>	Verifies the configuration.
Step 10	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Configuring IGMP Profiles

Follow these steps to create an IGMP profile:

This task is optional.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip igmp profile** *profile number*
4. **permit | deny**
5. **range ip** *multicast address*
6. **end**
7. **show ip igmp profile** *profile number*
8. **show running-config**
9. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	ip igmp profile <i>profile number</i> Example: Device(config)# ip igmp profile 3	Assigns a number to the profile you are configuring, and enters IGMP profile configuration mode. The profile number range is 1 to 4294967295. When you are in IGMP profile configuration mode, you can create the profile by using these commands: <ul style="list-style-type: none"> • deny—Specifies that matching addresses are denied; this is the default. • exit—Exits from igmp-profile configuration mode. • no—Negates a command or returns to its defaults. • permit—Specifies that matching addresses are permitted. • range—Specifies a range of IP addresses for the profile. You can enter a single IP address or a range with a start and an end address. The default is for the device to have no IGMP profiles configured. Note To delete a profile, use the no ip igmp profile <i>profile number</i> global configuration command.
Step 4	permit deny Example: Device(config-igmp-profile)# permit	(Optional) Sets the action to permit or deny access to the IP multicast address. If no action is configured, the default for the profile is to deny access.
Step 5	range <i>ip multicast address</i> Example: Device(config-igmp-profile)# range 229.9.9.0	Enters the IP multicast address or range of IP multicast addresses to which access is being controlled. If entering a range, enter the low IP multicast address, a space, and the high IP multicast address. You can use the range command multiple times to enter multiple addresses or ranges of addresses.

	Command or Action	Purpose
		Note To delete an IP multicast address or range of IP multicast addresses, use the no range ip multicast address IGMP profile configuration command.
Step 6	end Example: Device(config)# end	Returns to privileged EXEC mode.
Step 7	show ip igmp profile <i>profile number</i> Example: Device# show ip igmp profile 3	Verifies the profile configuration.
Step 8	show running-config Example: Device# show running-config	Verifies your entries.
Step 9	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Applying IGMP Profiles

To control access as defined in an IGMP profile, you have to apply the profile to the appropriate interfaces. You can apply IGMP profiles only to Layer 2 access ports; you cannot apply IGMP profiles to routed ports or SVIs. You cannot apply profiles to ports that belong to an EtherChannel port group. You can apply a profile to multiple interfaces, but each interface can have only one profile applied to it.

Follow these steps to apply an IGMP profile to a switch port:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **ip igmp filter** *profile number*
5. **end**
6. **show running-config**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> <code>enable</code>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode.
Step 3	interface <i>interface-id</i> Example: Device(config)# <code>interface gigabitethernet1/0/1</code>	Specifies the physical interface, and enters interface configuration mode. The interface must be a Layer 2 port that does not belong to an EtherChannel port group.
Step 4	ip igmp filter <i>profile number</i> Example: Device(config-if)# <code>ip igmp filter 321</code>	Applies the specified IGMP profile to the interface. The range is 1 to 4294967295. Note To remove a profile from an interface, use the no ip igmp filter <i>profile number</i> interface configuration command.
Step 5	end Example: Device(config-if)# <code>end</code>	Returns to privileged EXEC mode.
Step 6	show running-config Example: Device# <code>show running-config</code>	Verifies your entries.
Step 7	copy running-config startup-config Example: Device# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Setting the Maximum Number of IGMP Groups

Follow these steps to set the maximum number of IGMP groups that a Layer 2 interface can join:

Before you begin

This restriction can be applied to Layer 2 ports only; you cannot set a maximum number of IGMP groups on routed ports or SVIs. You also can use this command on a logical EtherChannel interface but cannot use it on ports that belong to an EtherChannel port group.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **ip igmp max-groups** *number*
5. **end**
6. **show running-config interface** *interface-id*
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	interface <i>interface-id</i> Example: Device(config)# interface gigabitethernet1/0/2	Specifies the interface to be configured, and enters interface configuration mode. The interface can be a Layer 2 port that does not belong to an EtherChannel group or a EtherChannel interface.
Step 4	ip igmp max-groups <i>number</i> Example: Device(config-if)# ip igmp max-groups 20	Sets the maximum number of IGMP groups that the interface can join. The range is 0 to 4294967294. The default is to have no maximum set. Note To remove the maximum group limitation and return to the default of no maximum, use the no ip igmp max-groups interface configuration command.
Step 5	end Example:	Returns to privileged EXEC mode.

	Command or Action	Purpose
	Device(config)# end	
Step 6	show running-config interface <i>interface-id</i> Example: Device# interface gigabitethernet1/0/1	Verifies your entries.
Step 7	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Configuring the IGMP Throttling Action

After you set the maximum number of IGMP groups that a Layer 2 interface can join, you can configure an interface to replace the existing group with the new group for which the IGMP report was received.

Follow these steps to configure the throttling action when the maximum number of entries is in the forwarding table:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface *interface-id***
4. **ip igmp max-groups action {deny | replace}**
5. **end**
6. **show running-config interface *interface-id***
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.

	Command or Action	Purpose
Step 3	interface <i>interface-id</i> Example: Device(config) # interface gigabitethernet1/0/1	Specifies the physical interface to be configured, and enters interface configuration mode. The interface can be a Layer 2 port that does not belong to an EtherChannel group or an EtherChannel interface. The interface cannot be a trunk port.
Step 4	ip igmp max-groups action {deny replace} Example: Device(config-if) # ip igmp max-groups action replace	When an interface receives an IGMP report and the maximum number of entries is in the forwarding table, specifies the action that the interface takes: <ul style="list-style-type: none"> • deny—Drops the report. If you configure this throttling action, the entries that were previously in the forwarding table are not removed but are aged out. After these entries are aged out and the maximum number of entries is in the forwarding table, the device drops the next IGMP report received on the interface. • replace—Replaces the existing group with the new group for which the IGMP report was received. If you configure this throttling action, the entries that were previously in the forwarding table are removed. When the maximum number of entries is in the forwarding table, the device replaces a randomly selected entry with the received IGMP report. To prevent the device from removing the forwarding-table entries, you can configure the IGMP throttling action before an interface adds entries to the forwarding table. Note To return to the default action of dropping the report, use the no ip igmp max-groups action interface configuration command.
Step 5	end Example: Device(config) # end	Returns to privileged EXEC mode.
Step 6	show running-config interface <i>interface-id</i> Example: Device# show running-config interface gigabitethernet1/0/1	Verifies your entries.
Step 7	copy running-config startup-config Example:	(Optional) Saves your entries in the configuration file.

Command or Action	Purpose
Device# <code>copy running-config startup-config</code>	

Monitoring IGMP Snooping and MVR

Monitoring IGMP Snooping Information

You can display IGMP snooping information for dynamically learned and statically configured router ports and VLAN interfaces. You can also display MAC address multicast entries for a VLAN configured for IGMP snooping.

Table 25: Commands for Displaying IGMP Snooping Information

Command	Purpose
<code>show ip igmp snooping [vlan <i>vlan-id</i> [detail]]</code>	Displays the snooping configuration information for all VLANs on the device or for a specified VLAN. (Optional) Enter vlan <i>vlan-id</i> to display information for a single VLAN. The VLAN ID range is 1 to 1001 and 1006 to 4094.
<code>show ip igmp snooping groups [count [dynamic [count] user [count]]]</code>	Displays multicast table information for the device or about a specific parameter: <ul style="list-style-type: none"> • count—Displays the total number of entries for the specified command options instead of the actual entries. • dynamic—Displays entries learned through IGMP snooping. • user—Displays only the user-configured multicast entries.
<code>show ip igmp snooping groups vlan <i>vlan-id</i> [<i>ip_address</i> count dynamic [count] user[count]]</code>	Displays multicast table information for a multicast VLAN or about a specific parameter for the VLAN: <ul style="list-style-type: none"> • <i>vlan-id</i>—The VLAN ID range is 1 to 1001 and 1006 to 4094. • count—Displays the total number of entries for the specified command options instead of the actual entries. • dynamic—Displays entries learned through IGMP snooping. • <i>ip_address</i>—Displays characteristics of the multicast group with the specified group IP address. • user—Displays only the user-configured multicast entries.

Command	Purpose
show ip igmp snooping mrouter [vlan <i>vlan-id</i>]	<p>Displays information on dynamically learned and manually configured multicast router interfaces.</p> <p>Note When you enable IGMP snooping, the device automatically learns the interface to which a multicast router is connected. These are dynamically learned interfaces.</p> <p>(Optional) Enter the vlan <i>vlan-id</i> to display information for a particular VLAN.</p>
show ip igmp snooping querier [vlan <i>vlan-id</i>] detail	Displays information about the IP address and receiving port of the most-recently received IGMP query message in the VLAN and the configuration and operational state of the IGMP snooping querier in the VLAN.

Monitoring MVR

You can monitor MVR for the switch or for a specified interface by displaying the following MVR information.

Table 26: Commands for Displaying MVR Information

Command	Purpose
show mvr	Displays MVR status and values for the switch—whether MVR is enabled or disabled, the multicast VLAN, the maximum (256) and current (0 through 256) number of multicast groups, the query response time, and the MVR mode.

Command	Purpose
show mvr interface [<i>interface-id</i>] [members [<i>vlan vlan-id</i>]]	<p>Displays all MVR interfaces and their MVR configurations.</p> <p>When a specific interface is entered, displays this information:</p> <ul style="list-style-type: none"> • Type—Receiver or Source • Status—One of these: <ul style="list-style-type: none"> • Active means the port is part of a VLAN. • Up/Down means that the port is forwarding or nonforwarding. • Inactive means that the port is not part of any VLAN. • Immediate Leave—Enabled or Disabled <p>If the members keyword is entered, displays all multicast group members on this port or, if a VLAN identification is entered, all multicast group members on the VLAN. The VLAN ID range is 1 to 1001 and 1006 to 4094.</p>
show mvr members [<i>ip-address</i>]	Displays all receiver and source ports that are members of any IP multicast group or the specified IP multicast group IP address.

Monitoring IGMP Filtering and Throttling Configuration

You can display IGMP profile characteristics, and you can display the IGMP profile and maximum group configuration for all interfaces on the device or for a specified interface. You can also display the IGMP throttling configuration for all interfaces on the device or for a specified interface.

Table 27: Commands for Displaying IGMP Filtering and Throttling Configuration

Command	Purpose
show ip igmp profile [<i>profile number</i>]	Displays the specified IGMP profile or all the IGMP profiles defined on the device.
show running-config [interface <i>interface-id</i>]	Displays the configuration of the specified interface or the configuration of all interfaces on the device, including (if configured) the maximum number of IGMP groups to which an interface can belong and the IGMP profile applied to the interface.

Configuration Examples for IGMP Snooping and MVR

Example: Configuring IGMP Snooping Using CGMP Packets

This example shows how to configure IGMP snooping to use CGMP packets as the learning method:

```
Device# configure terminal
Device(config)# ip igmp snooping vlan 1 mrouter learn cgmp
Device(config)# end
```

Example: Enabling a Static Connection to a Multicast Router

This example shows how to enable a static connection to a multicast router:

```
Device configure terminal
Device ip igmp snooping vlan 200 mrouter interface gigabitethernet1/0/2
Device end
```

Example: Configuring a Host Statically to Join a Group

This example shows how to statically configure a host on a port:

```
Device# configure terminal
Device# ip igmp snooping vlan 105 static 0100.1212.0000 interface gigabitethernet1/0/1
Device# end
```

Example: Enabling IGMP Immediate Leave

This example shows how to enable IGMP Immediate Leave on VLAN 130:

```
Device# configure terminal
Device(config)# ip igmp snooping vlan 130 immediate-leave
Device(config)# end
```

Example: Setting the IGMP Snooping Querier Source Address

This example shows how to set the IGMP snooping querier source address to 10.0.0.64:

```
Device# configure terminal
Device(config)# ip igmp snooping querier 10.0.0.64
Device(config)# end
```

Example: Setting the IGMP Snooping Querier Maximum Response Time

This example shows how to set the IGMP snooping querier maximum response time to 25 seconds:

```
Device# configure terminal
Device(config)# ip igmp snooping querier query-interval 25
Device(config)# end
```

Example: Setting the IGMP Snooping Querier Timeout

This example shows how to set the IGMP snooping querier timeout to 60 seconds:

```
Device# configure terminal
Device(config)# ip igmp snooping querier timeout expiry 60
Device(config)# end
```

Example: Setting the IGMP Snooping Querier Feature

This example shows how to set the IGMP snooping querier feature to Version 2:

```
Device# configure terminal
Device(config)# no ip igmp snooping querier version 2
Device(config)# end
```

Example: Configuring IGMP Profiles

This example shows how to create IGMP profile 4 allowing access to the single IP multicast address and how to verify the configuration. If the action was to deny (the default), it would not appear in the **show ip igmp profile** output display.

```
Device(config)# ip igmp profile 4
Device(config-igmp-profile)# permit
Device(config-igmp-profile)# range 229.9.9.0
Device(config-igmp-profile)# end
Device# show ip igmp profile 4
IGMP Profile 4
    permit
    range 229.9.9.0 229.9.9.0
```

Example: Applying IGMP Profile

This example shows how to apply IGMP profile 4 to a port:

```
Device(config)# interface gigabitethernet1/0/2
Device(config-if)# ip igmp filter 4
Device(config-if)# end
```

Example: Setting the Maximum Number of IGMP Groups

This example shows how to limit to 25 the number of IGMP groups that a port can join:

```
Device(config)# interface gigabitethernet1/0/2
Device(config-if)# ip igmp max-groups 25
Device(config-if)# end
```

Example: Configuring MVR Global Parameters

This example shows how to enable MVR, configure the group address, set the query time to 1 second (10 tenths), specify the MVR multicast VLAN as VLAN 22, and set the MVR mode as dynamic:

```
Device(config)# mvr
Device(config)# mvr group 228.1.23.4
Device(config)# mvr querytime 10
Device(config)# mvr vlan 22
Device(config)# mvr mode dynamic
Device(config)# end
```

Example: Configuring MVR Interfaces

This example shows how to configure a port as a receiver port, statically configure the port to receive multicast traffic sent to the multicast group address, configure Immediate Leave on the port, and verify the results:

```
Device(config)# mvr
Device(config)# interface gigabitethernet1/0/2
Device(config-if)# mvr type receiver
Device(config-if)# mvr vlan 22 group 228.1.23.4
Device(config-if)# mvr immediate
Device(config)# end
Device# show mvr interface
```

```
Port Type Status Immediate Leave
-----
Gi1/0/2 RECEIVER ACTIVE/DOWN ENABLED
```

Additional References

Related Documents

Related Topic	Document Title
For complete syntax and usage information for the commands used in this chapter.	<i>IP Multicast Command Reference, Cisco IOS Release 15.2(2)E (Catalyst 2960-XR Switch)</i>

Standards and RFCs

Standard/RFC	Title
RFC 1112	<i>Host Extensions for IP Multicasting</i>
RFC 2236	<i>Internet Group Management Protocol, Version 2</i>

MIBs

MIB	MIBs Link
All the supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
<p>The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies.</p> <p>To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds.</p> <p>Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.</p>	http://www.cisco.com/support

Feature History and Information for IGMP Snooping

Release	Modification
Cisco IOS Release 15.0(2)EX1	This feature was introduced.



CHAPTER 15

Configuring CGMP

- [Finding Feature Information](#), on page 183
- [Prerequisites for Configuring CGMP](#), on page 183
- [Restrictions for CGMP](#), on page 183
- [Information About CGMP](#), on page 184
- [Enabling CGMP Server Support](#), on page 184
- [Monitoring CGMP](#), on page 186
- [Additional References](#), on page 187
- [Feature History and Information for CGMP](#), on page 188

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see [Bug Search Tool](#) and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to <https://cfnng.cisco.com/>. An account on Cisco.com is not required.

Prerequisites for Configuring CGMP

The following are the prerequisites for configuring CGMP:

- When multiple Cisco CGMP-capable devices are connected to a switched network and the **ip cgmp proxy** command is needed, we recommend that all devices be configured with the same CGMP option and have precedence for becoming the IGMP querier over non-Cisco routers.

•

Restrictions for CGMP

The following are the restrictions for CGMP:

- CGMP is mutually exclusive with HSRPv1. You cannot enable CGMP leaving processing and HSRPv1 at the same time. However, you can enable CGMP and HSRPv2 at the same time.

Information About CGMP

Cisco Group Management Protocol or CGMP-server support is provided on the device; no client-side functionality is provided. The device serves as a CGMP server for devices that do not support IGMP snooping but have CGMP-client functionality.

CGMP is a protocol used on Cisco routers and multilayer devices connected to Layer 2 Catalyst devices to perform tasks similar to those performed by IGMP. CGMP permits Layer 2 group membership information to be communicated from the CGMP server to the device. The device can then learn on which interfaces multicast members reside instead of flooding multicast traffic to all device interfaces. (IGMP snooping is another method to constrain the flooding of multicast packets.)

CGMP is necessary because the Layer 2 device cannot distinguish between IP multicast data packets and IGMP report messages, which are both at the MAC level and are addressed to the same group address.

Enabling CGMP Server Support

When multiple Cisco CGMP-capable devices are connected to a switched network and you configure the **ip cgmp proxy** command, we recommend that all devices be configured with the same CGMP option and have precedence for becoming the IGMP querier over non-Cisco routers. Perform these steps to enable the CGMP server on the device interface:

This procedure is optional.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **ip cgmp** [**proxy** | **router-only**]
5. **end**
6. **show running-config**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. Enter your password, if prompted.
Step 2	configure terminal Example:	Enters global configuration mode.

	Command or Action	Purpose
	Device# <code>configure terminal</code>	
Step 3	interface <i>interface-id</i> Example: Device(config)# interface gigabitethernet 1/0/1	Specifies the interface that is connected to the Layer 2 Catalyst device, and enters interface configuration mode.
Step 4	ip cgmp [proxy router-only] Example: Device(config-if)# ip cgmp proxy	Enables CGMP on the interface. By default, CGMP is disabled on all interfaces. Enabling CGMP triggers a CGMP join message. Enable CGMP only on Layer 3 interfaces connected to Layer 2 Catalyst devices. (Optional) When you enter the proxy keyword, the CGMP proxy function is enabled. The proxy router advertises the existence of non-CGMP-capable routers by sending a CGMP join message with the non-CGMP-capable router MAC address and a group address of 0000.0000.0000. Note To perform CGMP proxy, the device must be the IGMP querier. If you configure the ip cgmp proxy command, you must manipulate the IP addresses so that the device is the IGMP querier, which might be the highest or lowest IP address, depending on which version of IGMP is running on the network. An IGMP Version 2 querier is selected based on the lowest IP address on the interface. An IGMP Version 1 querier is selected based on the multicast routing protocol used on the interface. Note To disable CGMP on the interface, use the no ip cgmp interface configuration command.
Step 5	end Example: Device(config-if)# end	Returns to privileged EXEC mode.
Step 6	show running-config Example: Device# show running-config	Verifies your entries.

	Command or Action	Purpose
Step 7	copy running-config startup-config Example: <pre>Device# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

What to do next

Verify the Layer 2 Catalyst device CGMP-client configuration. For more information, see the documentation that shipped with the product

Monitoring CGMP

You can display specific statistics, such as the contents of IP routing tables, caches, and databases.



Note This release does not support per-route statistics.

You can display information to learn resource usage and solve network problems. You can also display information about node reachability and discover the routing path that packets of your device are taking through the network.

You can use any of the privileged EXEC commands in the following table to display various routing statistics.

Table 28: Commands for Displaying System and Network Statistics

Command	Purpose
ping [<i>group-name</i> <i>group-address</i>]	Sends an ICMP Echo Request to a multicast group address.
show ip igmp groups [<i>group-name</i> <i>group-address</i> <i>type number</i>]	Displays the multicast groups that are directly connected to the switch and that were learned through IGMP.
show ip igmp interface [<i>type number</i>]	Displays multicast-related information about an interface.
show ip mcache [<i>group</i> [<i>source</i>]]	Displays the contents of the IP fast-switching cache.
show ip mpacket [<i>source-address</i> <i>name</i>] [<i>group-address</i> <i>name</i>] [detail]	Displays the contents of the circular cache-header buffer.
show ip mroute [<i>group-name</i> <i>group-address</i>] [<i>source</i>] [summary] [count] [active kbps]	Displays the contents of the IP multicast routing table.

Command	Purpose
show ip pim interface [<i>type number</i>] [count] [detail]	Displays information about interfaces configured for PIM. This command is available in all software images.
show ip pim neighbor [<i>type number</i>]	Lists the PIM neighbors discovered by the switch. This command is available in all software images.
show ip pim rp [<i>group-name</i> <i>group-address</i>]	Displays the RP routers associated with a sparse-mode multicast group. This command is available in all software images.
show ip rpf { <i>source-address</i> <i>name</i> }	Displays how the switch is doing Reverse-Path Forwarding (that is, from the unicast routing table, DVMRP routing table, or static mroutes).
show ip sap [<i>group</i> <i>session-name</i> detail]	Displays the Session Announcement Protocol (SAP) Version 2 cache.

Additional References

Related Documents

Related Topic	Document Title
For complete syntax and usage information for the commands used in this chapter.	<i>IP Multicast Command Reference, Cisco IOS Release 15.2(2)E (Catalyst 2960-XR Switch)</i>
Cisco IOS IP SLAs commands	Cisco IOS IP Multicast Command Reference

Standards and RFCs

Standard/RFC	Title
—	—

MIBs

MIB	MIBs Link
All the supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
<p>The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies.</p> <p>To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds.</p> <p>Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.</p>	http://www.cisco.com/support

Feature History and Information for CGMP

Release	Modification
Cisco IOS Release 15.0(2)EX1	This feature was introduced.



CHAPTER 16

Configuring Protocol Independent Multicast (PIM)

- [Prerequisites for PIM, on page 189](#)
- [Restrictions for PIM, on page 190](#)
- [Information About PIM, on page 192](#)
- [How to Configure PIM, on page 205](#)
- [Monitoring and Troubleshooting PIM, on page 237](#)
- [Configuration Examples for PIM, on page 239](#)
- [Additional References, on page 241](#)
- [Feature History and Information for PIM, on page 243](#)

Prerequisites for PIM

- Before you begin the PIM configuration process, decide which PIM mode to use. This is based on the applications you intend to support on your network. Use the following guidelines:
 - In general, if the application is one-to-many or many-to-many in nature, then PIM-SM can be used successfully.
 - For optimal one-to-many application performance, SSM is appropriate but requires IGMP version 3 support.
- Before you configure PIM stub routing, check that you have met these conditions:
 - You must have IP multicast routing configured on both the stub router and the central router. You must also have PIM mode (dense-mode, sparse-mode, or sparse-dense-mode) configured on the uplink interface of the stub router.
 - You must also configure Enhanced Interior Gateway Routing Protocol (EIGRP) stub routing on the device.
 - The PIM stub router does not route the transit traffic between the distribution routers. Unicast (EIGRP) stub routing enforces this behavior. You must configure unicast stub routing to assist the PIM stub router behavior.

Restrictions for PIM

PIMv1 and PIMv2 Interoperability

To avoid misconfiguring multicast routing on your device, review the information in this section.

The Cisco PIMv2 implementation provides interoperability and transition between Version 1 and Version 2, although there might be some minor problems.

You can upgrade to PIMv2 incrementally. PIM Versions 1 and 2 can be configured on different routers and multilayer devices within one network. Internally, all routers and multilayer devices on a shared media network must run the same PIM version. Therefore, if a PIMv2 device detects a PIMv1 device, the Version 2 device downgrades itself to Version 1 until all Version 1 devices have been shut down or upgraded.

PIMv2 uses the BSR to discover and announce RP-set information for each group prefix to all the routers and multilayer devices in a PIM domain. PIMv1, together with the Auto-RP feature, can perform the same tasks as the PIMv2 BSR. However, Auto-RP is a standalone protocol, separate from PIMv1, and is a proprietary Cisco protocol. PIMv2 is a standards track protocol in the IETF.



Note We recommend that you use PIMv2. The BSR function interoperates with Auto-RP on Cisco routers and multilayer devices.

When PIMv2 devices interoperate with PIMv1 devices, Auto-RP should have already been deployed. A PIMv2 BSR that is also an Auto-RP mapping agent automatically advertises the RP elected by Auto-RP. That is, Auto-RP sets its single RP on every router or multilayer device in the group. Not all routers and devices in the domain use the PIMv2 hash function to select multiple RPs.

Sparse-mode groups in a mixed PIMv1 and PIMv2 region are possible because the Auto-RP feature in PIMv1 interoperates with the PIMv2 RP feature. Although all PIMv2 devices can also use PIMv1, we recommend that the RPs be upgraded to PIMv2. To ease the transition to PIMv2, we recommend:

- Using Auto-RP throughout the region.
- Configuring sparse-dense mode throughout the region.

If Auto-RP is not already configured in the PIMv1 regions, configure Auto-RP.

Restrictions for Configuring PIM Stub Routing

- To use the PIM stub routing feature, the device or active stack can be running the IP base image. The IP base image contains only PIM stub routing. The IP services image contains complete multicast routing.
- Only directly connected multicast (IGMP) receivers and sources are allowed in the Layer 2 access domains. The PIM protocol is not supported in access domains.
- In a network using PIM stub routing, the only allowable route for IP traffic to the user is through a device that is configured with PIM stub routing.
- The redundant PIM stub router topology is not supported. Only the nonredundant access router topology is supported by the PIM stub feature.

Restrictions for Configuring Auto-RP and BSR

Take into consideration your network configuration, and the following restrictions when configuring Auto-RP and BSR:

Restrictions for Configuring Auto-RP

The following are restrictions for configuring Auto-RP (if used in your network configuration):

- If you configure PIM in sparse mode or sparse-dense mode and do not configure Auto-RP, you must manually configure an RP.
- If routed interfaces are configured in sparse mode, Auto-RP can still be used if all devices are configured with a manual RP address for the Auto-RP groups.
- If routed interfaces are configured in sparse mode and you enter the **ip pim autorp listener** global configuration command, Auto-RP can still be used even if all devices are not configured with a manual RP address for the Auto-RP groups.

Restrictions for Configuring BSR

The following are the restrictions for configuring BSR (if used in your network configuration):

- Configure the candidate BSRs as the RP-mapping agents for Auto-RP.
- For group prefixes advertised through Auto-RP, the PIMv2 BSR mechanism should not advertise a subrange of these group prefixes served by a different set of RPs. In a mixed PIMv1 and PIMv2 domain, have backup RPs serve the same group prefixes. This prevents the PIMv2 DRs from selecting a different RP from those PIMv1 DRs, due to the longest match lookup in the RP-mapping database.

Restrictions and Guidelines for Configuring Auto-RP and BSR

The following are restrictions for configuring Auto-RP and BSR (if used in your network configuration):

- If your network is all Cisco routers and multilayer devices, you can use either Auto-RP or BSR.
- If you have non-Cisco routers in your network, you must use BSR.
- If you have Cisco PIMv1 and PIMv2 routers and multilayer devices and non-Cisco routers, you must use both Auto-RP and BSR. If your network includes routers from other vendors, configure the Auto-RP mapping agent and the BSR on a Cisco PIMv2 device. Ensure that no PIMv1 device is located in the path a between the BSR and a non-Cisco PIMv2 device.



Note There are two approaches to using PIMv2. You can use Version 2 exclusively in your network or migrate to Version 2 by employing a mixed PIM version environment.

- Because bootstrap messages are sent hop-by-hop, a PIMv1 device prevents these messages from reaching all routers and multilayer devices in your network. Therefore, if your network has a PIMv1 device in it and only Cisco routers and multilayer devices, it is best to use Auto-RP.

- If you have a network that includes non-Cisco routers, configure the Auto-RP mapping agent and the BSR on a Cisco PIMv2 router or multilayer device. Ensure that no PIMv1 device is on the path between the BSR and a non-Cisco PIMv2 router.
- If you have non-Cisco PIMv2 routers that need to interoperate with Cisco PIMv1 routers and multilayer devices, both Auto-RP and a BSR are required. We recommend that a Cisco PIMv2 device be both the Auto-RP mapping agent and the BSR.

Information About PIM

Protocol Independent Multicast Overview

The Protocol Independent Multicast (PIM) protocol maintains the current IP multicast service mode of receiver-initiated membership. PIM is not dependent on a specific unicast routing protocol; it is IP routing protocol independent and can leverage whichever unicast routing protocols are used to populate the unicast routing table, including Enhanced Interior Gateway Routing Protocol (EIGRP), Open Shortest Path First (OSPF), Border Gateway Protocol (BGP), and static routes. PIM uses unicast routing information to perform the multicast forwarding function.

Although PIM is called a multicast routing protocol, it actually uses the unicast routing table to perform the reverse path forwarding (RPF) check function instead of building up a completely independent multicast routing table. Unlike other routing protocols, PIM does not send and receive routing updates between routers.

PIM is defined in RFC 4601, Protocol Independent Multicast - Sparse Mode (PIM-SM)

PIM can operate in dense mode or sparse mode. The router can also handle both sparse groups and dense groups at the same time (sparse-dense mode). The mode determines how the router populates its multicast routing table and how the router forwards multicast packets it receives from its directly connected LANs.

For information about PIM forwarding (interface) modes, see the following sections:

PIM Dense Mode

PIM dense mode (PIM-DM) uses a push model to flood multicast traffic to every corner of the network. This push model is a method for delivering data to the receivers without the receivers requesting the data. This method is efficient in certain deployments in which there are active receivers on every subnet in the network.

In dense mode, a router assumes that all other routers want to forward multicast packets for a group. If a router receives a multicast packet and has no directly connected members or PIM neighbors present, a prune message is sent back to the source. Subsequent multicast packets are not flooded to this router on this pruned branch. PIM builds source-based multicast distribution trees.

PIM-DM initially floods multicast traffic throughout the network. Routers that have no downstream neighbors prune back the unwanted traffic. This process repeats every 3 minutes.

Routers accumulate state information by receiving data streams through the flood and prune mechanism. These data streams contain the source and group information so that downstream routers can build up their multicast forwarding table. PIM-DM supports only source trees--that is, (S,G) entries--and cannot be used to build a shared distribution tree.



Note Dense mode is not often used and its use is not recommended. For this reason it is not specified in the configuration tasks in related modules.

PIM Sparse Mode

PIM sparse mode (PIM-SM) uses a pull model to deliver multicast traffic. Only network segments with active receivers that have explicitly requested the data will receive the traffic.

Sparse mode interfaces are added to the multicast routing table only when periodic Join messages are received from downstream routers, or when a directly connected member is on the interface. When forwarding from a LAN, sparse mode operation occurs if an RP is known for the group. If so, the packets are encapsulated and sent toward the RP. When no RP is known, the packet is flooded in a dense mode fashion. If the multicast traffic from a specific source is sufficient, the first hop router of the receiver may send Join messages toward the source to build a source-based distribution tree.

PIM-SM distributes information about active sources by forwarding data packets on the shared tree. Because PIM-SM uses shared trees (at least, initially), it requires the use of a rendezvous point (RP). The RP must be administratively configured in the network. See the [Rendezvous Points, on page 196](#) section for more information.

In sparse mode, a router assumes that other routers do not want to forward multicast packets for a group, unless there is an explicit request for the traffic. When hosts join a multicast group, the directly connected routers send PIM Join messages toward the RP. The RP keeps track of multicast groups. Hosts that send multicast packets are registered with the RP by the first hop router of that host. The RP then sends Join messages toward the source. At this point, packets are forwarded on a shared distribution tree. If the multicast traffic from a specific source is sufficient, the first hop router of the host may send Join messages toward the source to build a source-based distribution tree.

Sources register with the RP and then data is forwarded down the shared tree to the receivers. The edge routers learn about a particular source when they receive data packets on the shared tree from that source through the RP. The edge router then sends PIM (S,G) Join messages toward that source. Each router along the reverse path compares the unicast routing metric of the RP address to the metric of the source address. If the metric for the source address is better, it will forward a PIM (S,G) Join message toward the source. If the metric for the RP is the same or better, then the PIM (S,G) Join message will be sent in the same direction as the RP. In this case, the shared tree and the source tree would be considered congruent.

If the shared tree is not an optimal path between the source and the receiver, the routers dynamically create a source tree and stop traffic from flowing down the shared tree. This behavior is the default behavior in software. Network administrators can force traffic to stay on the shared tree by using the **ip pim spt-threshold infinity** command.

PIM-SM scales well to a network of any size, including those with WAN links. The explicit join mechanism prevents unwanted traffic from flooding the WAN links.

Sparse-Dense Mode

If you configure either sparse mode or dense mode on an interface, then sparseness or denseness is applied to the interface as a whole. However, some environments might require PIM to run in a single region in sparse mode for some groups and in dense mode for other groups.

An alternative to enabling only dense mode or only sparse mode is to enable sparse-dense mode. In this case, the interface is treated as dense mode if the group is in dense mode; the interface is treated in sparse mode if

the group is in sparse mode. You must have an RP if the interface is in sparse-dense mode and you want to treat the group as a sparse group.

If you configure sparse-dense mode, the idea of sparseness or denseness is applied to the groups for which the router is a member.

Another benefit of sparse-dense mode is that Auto-RP information can be distributed in a dense mode; yet, multicast groups for user groups can be used in a sparse mode manner. Therefore there is no need to configure a default RP at the leaf routers.

When an interface is treated in dense mode, it is populated in the outgoing interface list of a multicast routing table when either of the following conditions is true:

- Members or DVMRP neighbors are on the interface.
- There are PIM neighbors and the group has not been pruned.

When an interface is treated in sparse mode, it is populated in the outgoing interface list of a multicast routing table when either of the following conditions is true:

- Members or DVMRP neighbors are on the interface.
- An explicit Join message has been received by a PIM neighbor on the interface.

PIM Versions

PIMv2 includes these improvements over PIMv1:

- A single, active rendezvous point (RP) exists per multicast group, with multiple backup RPs. This single RP compares to multiple active RPs for the same group in PIMv1.
- A bootstrap router (BSR) provides a fault-tolerant, automated RP discovery and distribution function that enables routers and multilayer devices to dynamically learn the group-to-RP mappings.
- Sparse mode and dense mode are properties of a group, as opposed to an interface.



Note We strongly recommend using sparse-dense mode as opposed to either sparse mode or dense mode only.

- PIM join and prune messages have more flexible encoding for multiple address families.
- A more flexible hello packet format replaces the query packet to encode current and future capability options.
- Register messages sent to an RP specify whether they are sent by a border router or a designated router.
- PIM packets are no longer inside IGMP packets; they are standalone packets.

PIM Stub Routing

The PIM stub routing feature, available in all of the device software images, reduces resource usage by moving routed traffic closer to the end user.



Note The IP Base image contains only PIM stub routing. The IP Services image contains complete multicast routing. On a switch running the IP Base image, if you try to configure a VLAN interface with PIM dense-mode, sparse-mode, or dense-sparse-mode, the configuration is not allowed.

The PIM stub routing feature supports multicast routing between the distribution layer and the access layer. It supports two types of PIM interfaces, uplink PIM interfaces, and PIM passive interfaces. A routed interface configured with the PIM passive mode does not pass or forward PIM control traffic, it only passes and forwards IGMP traffic.

In a network using PIM stub routing, the only allowable route for IP traffic to the user is through a device that is configured with PIM stub routing. PIM passive interfaces are connected to Layer 2 access domains, such as VLANs, or to interfaces that are connected to other Layer 2 devices. Only directly connected multicast (IGMP) receivers and sources are allowed in the Layer 2 access domains. The PIM passive interfaces do not send or process any received PIM control packets.

When using PIM stub routing, you should configure the distribution and remote routers to use IP multicast routing and configure only the device as a PIM stub router. The device does not route transit traffic between distribution routers. You also need to configure a routed uplink port on the device. The device uplink port cannot be used with SVIs. If you need PIM for an SVI uplink port, you should upgrade to the IP Services feature set.



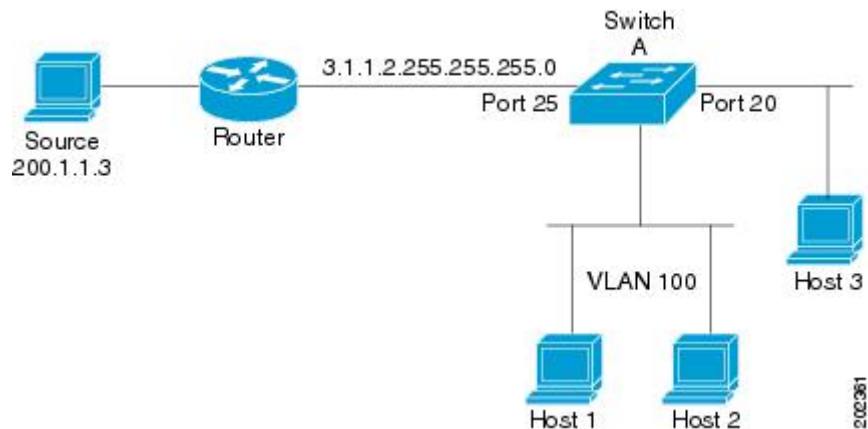
Note You must also configure EIGRP stub routing when configuring PIM stub routing on the device

The redundant PIM stub router topology is not supported. The redundant topology exists when there is more than one PIM router forwarding multicast traffic to a single access domain. PIM messages are blocked, and the PIM asset and designated router election mechanisms are not supported on the PIM passive interfaces. Only the nonredundant access router topology is supported by the PIM stub feature. By using a nonredundant topology, the PIM passive interface assumes that it is the only interface and designated router on that access domain.

The PIM stub feature is enforced in the IP Base image. If you upgrade to a higher software version, the PIM stub configuration remains until you reconfigure the interfaces.

Figure 10: PIM Stub Router Configuration

In the following figure, the Device A routed uplink port 25 is connected to the router and PIM stub routing is enabled on the VLAN 100 interfaces and on Host 3. This configuration allows the directly connected hosts to receive traffic from multicast source 200.1.1.3.



IGMP Helper

PIM stub routing moves routed traffic closer to the end user and reduces network traffic. You can also reduce traffic by configuring a stub router (switch) with the IGMP helper feature.

You can configure a stub router (switch) with the `ip igmp helper-address ip-address` interface configuration command to enable the switch to send reports to the next-hop interface. Hosts that are not directly connected to a downstream router can then join a multicast group sourced from an upstream network. The IGMP packets from a host wanting to join a multicast stream are forwarded upstream to the next-hop device when this feature is configured. When the upstream central router receives the helper IGMP reports or leaves, it adds or removes the interfaces from its outgoing interface list for that group.

Rendezvous Points

A rendezvous point (RP) is a role that a device performs when operating in Protocol Independent Multicast (PIM) Sparse Mode (SM). An RP is required only in networks running PIM SM. In the PIM-SM model, only network segments with active receivers that have explicitly requested multicast data will be forwarded the traffic.

This method of delivering multicast data is in contrast to PIM Dense Mode (PIM DM). In PIM DM, multicast traffic is initially flooded to all segments of the network. Routers that have no downstream neighbors or directly connected receivers prune back the unwanted traffic.

An RP acts as the meeting place for sources and receivers of multicast data. In a PIM-SM network, sources must send their traffic to the RP. This traffic is then forwarded to receivers down a shared distribution tree. By default, when the first hop device of the receiver learns about the source, it will send a Join message directly to the source, creating a source-based distribution tree from the source to the receiver. This source tree does not include the RP unless the RP is located within the shortest path between the source and receiver.

In most cases, the placement of the RP in the network is not a complex decision. By default, the RP is needed only to start new sessions with sources and receivers. Consequently, the RP experiences little overhead from traffic flow or processing. In PIM version 2, the RP performs less processing than in PIM version 1 because sources must only periodically register with the RP to create state.

Auto-RP

In the first version of PIM-SM, all leaf routers (routers directly connected to sources or receivers) were required to be manually configured with the IP address of the RP. This type of configuration is also known as static

RP configuration. Configuring static RPs is relatively easy in a small network, but it can be laborious in a large, complex network.

Following the introduction of PIM-SM version 1, Cisco implemented a version of PIM-SM with the Auto-RP feature. Auto-RP automates the distribution of group-to-RP mappings in a PIM network. Auto-RP has the following benefits:

- Configuring the use of multiple RPs within a network to serve different groups is easy.
- Auto-RP allows load splitting among different RPs and arrangement of RPs according to the location of group participants.
- Auto-RP avoids inconsistent, manual RP configurations that can cause connectivity problems.

Multiple RPs can be used to serve different group ranges or serve as backups to each other. For Auto-RP to work, a router must be designated as an RP-mapping agent, which receives the RP-announcement messages from the RPs and arbitrates conflicts. The RP-mapping agent then sends the consistent group-to-RP mappings to all other routers. Thus, all routers automatically discover which RP to use for the groups they support.



Note If you configure PIM in sparse mode or sparse-dense mode and do not configure Auto-RP, you must statically configure an RP.



Note If router interfaces are configured in sparse mode, Auto-RP can still be used if all routers are configured with a static RP address for the Auto-RP groups.

To make Auto-RP work, a router must be designated as an RP mapping agent, which receives the RP announcement messages from the RPs and arbitrates conflicts. The RP mapping agent then sends the consistent group-to-RP mappings to all other routers by dense mode flooding. Thus, all routers automatically discover which RP to use for the groups they support. The Internet Assigned Numbers Authority (IANA) has assigned two group addresses, 224.0.1.39 and 224.0.1.40, for Auto-RP. One advantage of Auto-RP is that any change to the RP designation must be configured only on the routers that are RPs and not on the leaf routers. Another advantage of Auto-RP is that it offers the ability to scope the RP address within a domain. Scoping can be achieved by defining the time-to-live (TTL) value allowed for the Auto-RP advertisements.

Each method for configuring an RP has its own strengths, weaknesses, and level of complexity. In conventional IP multicast network scenarios, we recommend using Auto-RP to configure RPs because it is easy to configure, well-tested, and stable. The alternative ways to configure an RP are static RP, Auto-RP, and bootstrap router.

Sparse-Dense Mode for Auto-RP

A prerequisite of Auto-RP is that all interfaces must be configured in sparse-dense mode using the **ip pim sparse-dense-mode** interface configuration command. An interface configured in sparse-dense mode is treated in either sparse mode or dense mode of operation, depending on which mode the multicast group operates. If a multicast group has a known RP, the interface is treated in sparse mode. If a group has no known RP, by default the interface is treated in dense mode and data will be flooded over this interface. (You can prevent dense-mode fallback; see the module “Configuring Basic IP Multicast.”)

To successfully implement Auto-RP and prevent any groups other than 224.0.1.39 and 224.0.1.40 from operating in dense mode, we recommend configuring a “sink RP” (also known as “RP of last resort”). A sink RP is a statically configured RP that may or may not actually exist in the network. Configuring a sink RP

does not interfere with Auto-RP operation because, by default, Auto-RP messages supersede static RP configurations. We recommend configuring a sink RP for all possible multicast groups in your network, because it is possible for an unknown or unexpected source to become active. If no RP is configured to limit source registration, the group may revert to dense mode operation and be flooded with data.

Bootstrap Router

Another RP selection model called bootstrap router (BSR) was introduced after Auto-RP in PIM-SM version 2. BSR performs similarly to Auto-RP in that it uses candidate routers for the RP function and for relaying the RP information for a group. RP information is distributed through BSR messages, which are carried within PIM messages. PIM messages are link-local multicast messages that travel from PIM router to PIM router. Because of this single hop method of disseminating RP information, TTL scoping cannot be used with BSR. A BSR performs similarly as an RP, except that it does not run the risk of reverting to dense mode operation, and it does not offer the ability to scope within a domain.

PIM Domain Border

As IP multicast becomes more widespread, the chance of one PIMv2 domain bordering another PIMv2 domain increases. Because two domains probably do not share the same set of RPs, BSR, candidate RPs, and candidate BSRs, you need to constrain PIMv2 BSR messages from flowing into or out of the domain. Allowing messages to leak across the domain borders could adversely affect the normal BSR election mechanism and elect a single BSR across all bordering domains and coningle candidate RP advertisements, resulting in the election of RPs in the wrong domain.

Multicast Forwarding

Forwarding of multicast traffic is accomplished by multicast-capable routers. These routers create distribution trees that control the path that IP multicast traffic takes through the network in order to deliver traffic to all receivers.

Multicast traffic flows from the source to the multicast group over a distribution tree that connects all of the sources to all of the receivers in the group. This tree may be shared by all sources (a shared tree) or a separate distribution tree can be built for each source (a source tree). The shared tree may be one-way or bidirectional.

Before describing the structure of source and shared trees, it is helpful to explain the notations that are used in multicast routing tables. These notations include the following:

- (S,G) = (unicast source for the multicast group G, multicast group G)
- (*,G) = (any source for the multicast group G, multicast group G)

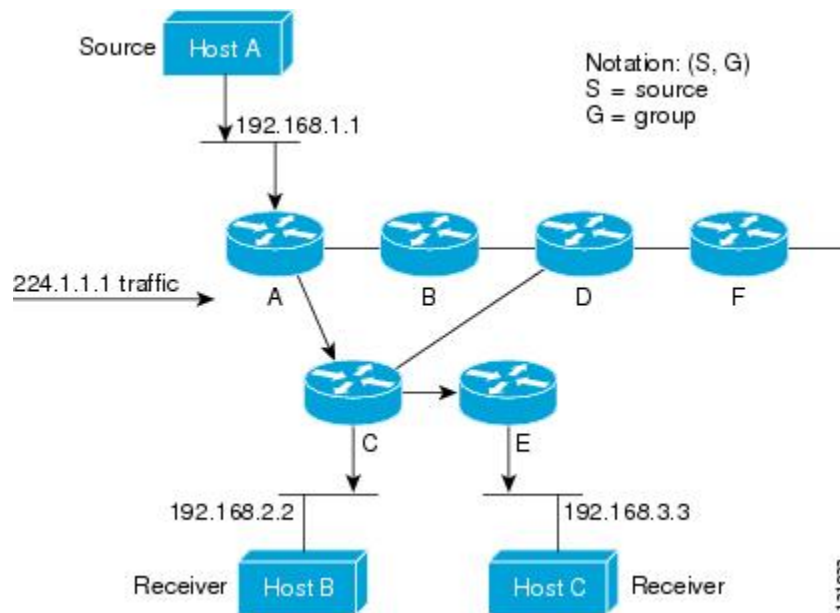
The notation of (S,G), pronounced “S comma G,” enumerates a shortest path tree where S is the IP address of the source and G is the multicast group address.

Shared trees are (*,G) and the source trees are (S,G) and always routed at the sources.

Multicast Distribution Source Tree

The simplest form of a multicast distribution tree is a source tree. A source tree has its root at the source host and has branches forming a spanning tree through the network to the receivers. Because this tree uses the shortest path through the network, it is also referred to as a shortest path tree (SPT).

The figure shows an example of an SPT for group 224.1.1.1 rooted at the source, Host A, and connecting two receivers, Hosts B and C.



Using standard notation, the SPT for the example shown in the figure would be (192.168.1.1, 224.1.1.1).

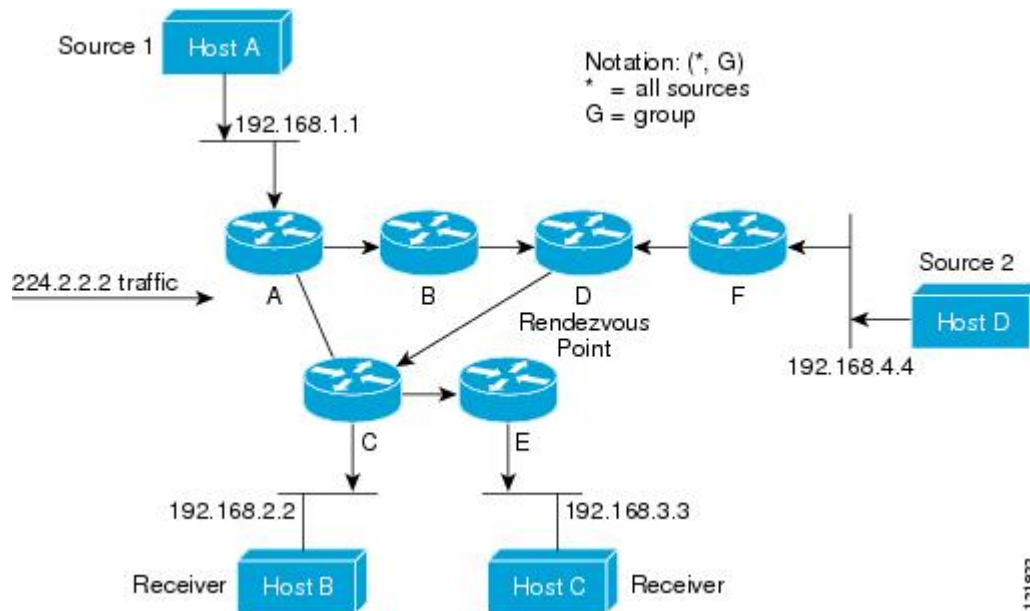
The (S,G) notation implies that a separate SPT exists for each individual source sending to each group--which is correct.

Multicast Distribution Shared Tree

Unlike source trees that have their root at the source, shared trees use a single common root placed at some chosen point in the network. This shared root is called a rendezvous point (RP).

The following figure shows a shared tree for the group 224.2.2.2 with the root located at Router D. This shared tree is unidirectional. Source traffic is sent towards the RP on a source tree. The traffic is then forwarded down the shared tree from the RP to reach all of the receivers (unless the receiver is located between the source and the RP, in which case it will be serviced directly).

Figure 11: Shared Tree



In this example, multicast traffic from the sources, Hosts A and D, travels to the root (Router D) and then down the shared tree to the two receivers, Hosts B and C. Because all sources in the multicast group use a common shared tree, a wildcard notation written as (*, G), pronounced “star comma G,” represents the tree. In this case, * means all sources, and G represents the multicast group. Therefore, the shared tree shown in the figure would be written as (*, 224.2.2.2).

Both source trees and shared trees are loop-free. Messages are replicated only where the tree branches. Members of multicast groups can join or leave at any time; therefore the distribution trees must be dynamically updated. When all the active receivers on a particular branch stop requesting the traffic for a particular multicast group, the routers prune that branch from the distribution tree and stop forwarding traffic down that branch. If one receiver on that branch becomes active and requests the multicast traffic, the router will dynamically modify the distribution tree and start forwarding traffic again.

Source Tree Advantage

Source trees have the advantage of creating the optimal path between the source and the receivers. This advantage guarantees the minimum amount of network latency for forwarding multicast traffic. However, this optimization comes at a cost. The routers must maintain path information for each source. In a network that has thousands of sources and thousands of groups, this overhead can quickly become a resource issue on the routers. Memory consumption from the size of the multicast routing table is a factor that network designers must take into consideration.

Shared Tree Advantage

Shared trees have the advantage of requiring the minimum amount of state in each router. This advantage lowers the overall memory requirements for a network that only allows shared trees. The disadvantage of shared trees is that under certain circumstances the paths between the source and receivers might not be the optimal paths, which might introduce some latency in packet delivery. For example, in the figure above the shortest path between Host A (source 1) and Host B (a receiver) would be Router A and Router C. Because we are using Router D as the root for a shared tree, the traffic must traverse Routers A, B, D and then C.

Network designers must carefully consider the placement of the rendezvous point (RP) when implementing a shared tree-only environment.

In unicast routing, traffic is routed through the network along a single path from the source to the destination host. A unicast router does not consider the source address; it considers only the destination address and how to forward the traffic toward that destination. The router scans through its routing table for the destination address and then forwards a single copy of the unicast packet out the correct interface in the direction of the destination.

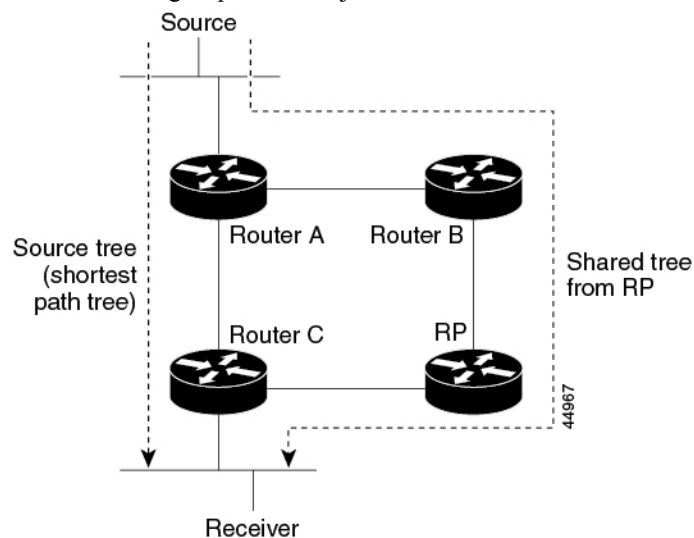
In multicast forwarding, the source is sending traffic to an arbitrary group of hosts that are represented by a multicast group address. The multicast router must determine which direction is the upstream direction (toward the source) and which one is the downstream direction (or directions) toward the receivers. If there are multiple downstream paths, the router replicates the packet and forwards it down the appropriate downstream paths (best unicast route metric)--which is not necessarily all paths. Forwarding multicast traffic away from the source, rather than to the receiver, is called Reverse Path Forwarding (RPF). RPF is described in the following section.

PIM Shared Tree and Source Tree

By default, members of a group receive data from senders to the group across a single data-distribution tree rooted at the RP.

Figure 12: Shared Tree and Source Tree (Shortest-Path Tree)

The following figure shows this type of shared-distribution tree. Data from senders is delivered to the RP for distribution to group members joined to the shared tree.



If the data rate warrants, leaf routers (routers without any downstream connections) on the shared tree can use the data distribution tree rooted at the source. This type of distribution tree is called a shortest-path tree or source tree. By default, the software devices to a source tree upon receiving the first data packet from a source.

This process describes the move from a shared tree to a source tree:

1. A receiver joins a group; leaf Router C sends a join message toward the RP.
2. The RP puts a link to Router C in its outgoing interface list.

3. A source sends data; Router A encapsulates the data in a register message and sends it to the RP.
4. The RP forwards the data down the shared tree to Router C and sends a join message toward the source. At this point, data might arrive twice at Router C, once encapsulated and once natively.
5. When data arrives natively (unencapsulated) at the RP, it sends a register-stop message to Router A.
6. By default, reception of the first data packet prompts Router C to send a join message toward the source.
7. When Router C receives data on (S, G), it sends a prune message for the source up the shared tree.
8. The RP deletes the link to Router C from the outgoing interface of (S, G). The RP triggers a prune message toward the source.

Join and prune messages are sent for sources and RPs. They are sent hop-by-hop and are processed by each PIM device along the path to the source or RP. Register and register-stop messages are not sent hop-by-hop. They are sent by the designated router that is directly connected to a source and are received by the RP for the group.

Multiple sources sending to groups use the shared tree. You can configure the PIM device to stay on the shared tree.

The change from shared to source tree happens when the first data packet arrives at the last-hop router. This change depends upon the threshold that is configured by using the **ip pim spt-threshold** global configuration command.

The shortest-path tree requires more memory than the shared tree but reduces delay. You may want to postpone its use. Instead of allowing the leaf router to immediately move to the shortest-path tree, you can specify that the traffic must first reach a threshold.

You can configure when a PIM leaf router should join the shortest-path tree for a specified group. If a source sends at a rate greater than or equal to the specified kbps rate, the multilayer switch triggers a PIM join message toward the source to construct a source tree (shortest-path tree). If the traffic rate from the source drops below the threshold value, the leaf router switches back to the shared tree and sends a prune message toward the source.

You can specify to which groups the shortest-path tree threshold applies by using a group list (a standard access list). If a value of 0 is specified or if the group list is not used, the threshold applies to all groups.

Reverse Path Forwarding

In unicast routing, traffic is routed through the network along a single path from the source to the destination host. A unicast router does not consider the source address; it considers only the destination address and how to forward the traffic toward that destination. The router scans through its routing table for the destination network and then forwards a single copy of the unicast packet out the correct interface in the direction of the destination.

In multicast forwarding, the source is sending traffic to an arbitrary group of hosts that are represented by a multicast group address. The multicast router must determine which direction is the upstream direction (toward the source) and which one is the downstream direction (or directions) toward the receivers. If there are multiple downstream paths, the router replicates the packet and forwards it down the appropriate downstream paths (best unicast route metric)--which is not necessarily all paths. Forwarding multicast traffic away from the source, rather than to the receiver, is called Reverse Path Forwarding (RPF). RPF is an algorithm used for forwarding multicast datagrams.

Protocol Independent Multicast (PIM) uses the unicast routing information to create a distribution tree along the reverse path from the receivers towards the source. The multicast routers then forward packets along the

distribution tree from the source to the receivers. RPF is a key concept in multicast forwarding. It enables routers to correctly forward multicast traffic down the distribution tree. RPF makes use of the existing unicast routing table to determine the upstream and downstream neighbors. A router will forward a multicast packet only if it is received on the upstream interface. This RPF check helps to guarantee that the distribution tree will be loop-free.

RPF Check

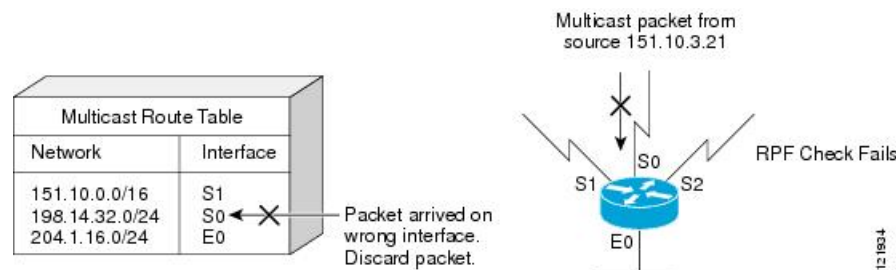
When a multicast packet arrives at a router, the router performs an RPF check on the packet. If the RPF check succeeds, the packet is forwarded. Otherwise, it is dropped.

For traffic flowing down a source tree, the RPF check procedure works as follows:

1. The router looks up the source address in the unicast routing table to determine if the packet has arrived on the interface that is on the reverse path back to the source.
2. If the packet has arrived on the interface leading back to the source, the RPF check succeeds and the packet is forwarded out the interfaces present in the outgoing interface list of a multicast routing table entry.
3. If the RPF check in Step 2 fails, the packet is dropped.

The figure shows an example of an unsuccessful RPF check.

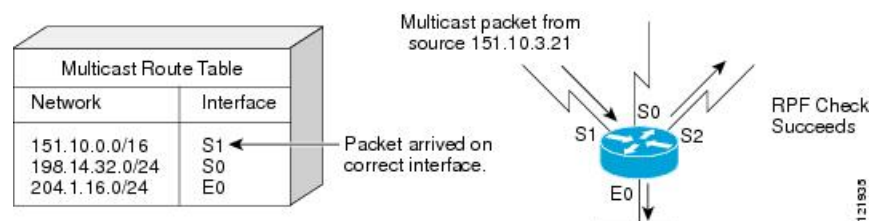
Figure 13: RPF Check Fails



As the figure illustrates, a multicast packet from source 151.10.3.21 is received on serial interface 0 (S0). A check of the unicast route table shows that S1 is the interface this router would use to forward unicast data to 151.10.3.21. Because the packet has arrived on interface S0, the packet is discarded.

The figure shows an example of a successful RPF check.

Figure 14: RPF Check Succeeds



In this example, the multicast packet has arrived on interface S1. The router refers to the unicast routing table and finds that S1 is the correct interface. The RPF check passes, and the packet is forwarded.

PIM uses both source trees and RP-rooted shared trees to forward datagrams. The RPF check is performed differently for each:

- If a PIM router or multilayer device has a source-tree state (that is, an (S, G) entry is present in the multicast routing table), it performs the RPF check against the IP address of the source of the multicast packet.
- If a PIM router or multilayer device has a shared-tree state (and no explicit source-tree state), it performs the RPF check on the RP address (which is known when members join the group).

DVMRP and dense-mode PIM use only source trees and use RPF.



Note DVMRP is not supported on the device.

Sparse-mode PIM uses the RPF lookup function to decide where it needs to send joins and prunes:

- (S, G) joins (which are source-tree states) are sent toward the source.
- (*,G) joins (which are shared-tree states) are sent toward the RP.

Default PIM Routing Configuration

This table displays the default PIM routing configuration for the device.

Table 29: Default Multicast Routing Configuration

Feature	Default Setting
Multicast routing	Disabled on all interfaces.
PIM version	Version 2.
PIM mode	No mode is defined.
PIM stub routing	None configured.
PIM RP address	None configured.
PIM domain border	Disabled.
PIM multicast boundary	None.
Candidate BSRs	Disabled.
Candidate RPs	Disabled.
Shortest-path tree threshold rate	0 kb/s.
PIM router query message interval	30 seconds.

How to Configure PIM

Enabling PIM Stub Routing

This procedure is optional.

SUMMARY STEPS

1. `enable`
2. `configure terminal`
3. `interface interface-id`
4. `ip pim passive`
5. `end`
6. `show ip pim interface`
7. `show running-config`
8. `copy running-config startup-config`

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> <code>enable</code>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode.
Step 3	interface interface-id Example: Device(config)# <code>interface gigabitethernet 1/0/1</code>	Specifies the interface on which you want to enable PIM stub routing, and enters interface configuration mode.
Step 4	ip pim passive Example: Device(config-if)# <code>ip pim passive</code>	Configures the PIM stub feature on the interface.
Step 5	end Example:	Returns to privileged EXEC mode.

	Command or Action	Purpose
	Device(config)# end	
Step 6	show ip pim interface Example: Device# show ip pim interface	(Optional) Displays the PIM stub that is enabled on each interface.
Step 7	show running-config Example: Device# show running-config	Verifies your entries.
Step 8	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Configuring a Rendezvous Point

You must have a rendezvous point (RP), if the interface is in sparse-dense mode and if you want to handle the group as a sparse group. You can use these methods:

- By manually assigning an RP to multicast groups.
- As a standalone, Cisco-proprietary protocol separate from PIMv1, which includes:
 - Setting up Auto-RP in a new internetwork
 - Adding Auto-RP to an existing sparse-mode cloud
 - Preventing join messages to false RPs
 - Filtering incoming RP announcement messages
- By using a standards track protocol in the Internet Engineering Task Force (IETF), which includes configuring PIMv2 BSR .



Note You can use Auto-RP, BSR, or a combination of both, depending on the PIM version that you are running and the types of routers in your network. For information about working with different PIM versions in your network, see the PIMv1 and PIMv2 Interoperability section.

Manually Assigning an RP to Multicast Groups

If the rendezvous point (RP) for a group is learned through a dynamic mechanism (such as Auto-RP or BSR), you need not perform this task for that RP.

Senders of multicast traffic announce their existence through register messages received from the source first-hop router (designated router) and forwarded to the RP. Receivers of multicast packets use RPs to join a multicast group by using explicit join messages.



Note RPs are not members of the multicast group; they serve as a *meeting place* for multicast sources and group members.

You can configure a single RP for multiple groups defined by an access list. If there is no RP configured for a group, the multilayer device responds to the group as dense and uses the dense-mode PIM techniques.

This procedure is optional.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip pim rp-address** *ip-address* [*access-list-number*] [**override**]
4. **access-list** *access-list-number* {**deny** | **permit**} *source* [*source-wildcard*]
5. **end**
6. **show running-config**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	ip pim rp-address <i>ip-address</i> [<i>access-list-number</i>] [override] Example: Device(config)# ip pim rp-address	Configures the address of a PIM RP. By default, no PIM RP address is configured. You must configure the IP address of RPs on all routers and multilayer devices (including the RP).

	Command or Action	Purpose
	<pre>10.1.1.1 20 override</pre>	<p>Note If there is no RP configured for a group, the device treats the group as dense, using the dense-mode PIM techniques.</p> <p>A PIM device can be an RP for more than one group. Only one RP address can be used at a time within a PIM domain. The access list conditions specify for which groups the device is an RP.</p> <ul style="list-style-type: none"> • For <i>ip-address</i>, enter the unicast address of the RP in dotted-decimal notation. • (Optional) For <i>access-list-number</i>, enter an IP standard access list number from 1 to 99. If no access list is configured, the RP is used for all groups. • (Optional) The override keyword indicates that if there is a conflict between the RP configured with this command and one learned by Auto-RP or BSR, the RP configured with this command prevails. <p>Note To remove an RP address, use the no ip pim rp-address ip-address [access-list-number] [override] global configuration command.</p>
Step 4	<p>access-list <i>access-list-number</i> {deny permit} <i>source</i> [<i>source-wildcard</i>]</p> <p>Example:</p> <pre>Device(config)# access-list 25 permit 10.5.0.1 255.224.0.0</pre>	<p>Creates a standard access list, repeating the command as many times as necessary.</p> <ul style="list-style-type: none"> • For <i>access-list-number</i>, enter the access list number specified in Step 2. • The deny keyword denies access if the conditions are matched. • The permit keyword permits access if the conditions are matched. • For <i>source</i>, enter the multicast group address for which the RP should be used. • (Optional) For <i>source-wildcard</i>, enter the wildcard bits in dotted decimal notation to be applied to the source. Place ones in the bit positions that you want to ignore. <p>The access list is always terminated by an implicit deny statement for everything.</p>
Step 5	<p>end</p> <p>Example:</p> <pre>Device(config)# end</pre>	<p>Returns to privileged EXEC mode.</p>

	Command or Action	Purpose
Step 6	show running-config Example: Device# <code>show running-config</code>	Verifies your entries.
Step 7	copy running-config startup-config Example: Device# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Setting Up Auto-RP in a New Internetwork

If you are setting up Auto-RP in a new internetwork, you do not need a default RP because you configure all the interfaces for sparse-dense mode.



Note Omit Step 3 in the following procedure, if you want to configure a PIM router as the RP for the local group.

SUMMARY STEPS

1. `enable`
2. `show running-config`
3. `configure terminal`
4. `ip pim send-rp-announce interface-id scope ttl group-list access-list-number interval seconds`
5. `access-list access-list-number {deny | permit} source [source-wildcard]`
6. `ip pim send-rp-discovery scope ttl`
7. `end`
8. `show running-config`
9. `show ip pim rp mapping`
10. `show ip pim rp`
11. `copy running-config startup-config`

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> <code>enable</code>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	show running-config Example:	Verifies that a default RP is already configured on all PIM devices and the RP in the sparse-mode network. It was

	Command or Action	Purpose
	<pre>Device# show running-config</pre>	<p>previously configured with the ip pim rp-address global configuration command.</p> <p>Note This step is not required for sparse-dense-mode environments.</p> <p>The selected RP should have good connectivity and be available across the network. Use this RP for the global groups (for example, 224.x.x.x and other global groups). Do not reconfigure the group address range that this RP serves. RPs dynamically discovered through Auto-RP take precedence over statically configured RPs. Assume that it is desirable to use a second RP for the local groups.</p>
Step 3	<p>configure terminal</p> <p>Example:</p> <pre>Device# configure terminal</pre>	Enters global configuration mode.
Step 4	<p>ip pim send-rp-announce <i>interface-id</i> scope <i>ttl</i> group-list <i>access-list-number</i> interval <i>seconds</i></p> <p>Example:</p> <pre>Device(config)# ip pim send-rp-announce gigabitethernet 1/0/5 scope 20 group-list 10 interval 120</pre>	<p>Configures another PIM device to be the candidate RP for local groups.</p> <ul style="list-style-type: none"> For <i>interface-id</i>, enter the interface type and number that identifies the RP address. Valid interfaces include physical ports, port channels, and VLANs. For scope <i>ttl</i>, specify the time-to-live value in hops. Enter a hop count that is high enough so that the RP-announce messages reach all mapping agents in the network. There is no default setting. The range is 1 to 255. For group-list <i>access-list-number</i>, enter an IP standard access list number from 1 to 99. If no access list is configured, the RP is used for all groups. For interval <i>seconds</i>, specify how often the announcement messages must be sent. The default is 60 seconds. The range is 1 to 16383.
Step 5	<p>access-list <i>access-list-number</i> {deny permit} <i>source</i> [<i>source-wildcard</i>]</p> <p>Example:</p> <pre>Device(config)# access-list 10 permit 10.10.0.0</pre>	<p>Creates a standard access list, repeating the command as many times as necessary.</p> <ul style="list-style-type: none"> For <i>access-list-number</i>, enter the access list number specified in Step 3. The deny keyword denies access if the conditions are matched. The permit keyword permits access if the conditions are matched.

	Command or Action	Purpose
		<ul style="list-style-type: none"> For <i>source</i>, enter the multicast group address range for which the RP should be used. (Optional) For <i>source-wildcard</i>, enter the wildcard bits in dotted decimal notation to be applied to the source. Place ones in the bit positions that you want to ignore. <p>Note Recall that the access list is always terminated by an implicit deny statement for everything.</p>
Step 6	<p>ip pim send-rp-discovery scope ttl</p> <p>Example:</p> <pre>Device(config)# ip pim send-rp-discovery scope 50</pre>	<p>Finds a device whose connectivity is not likely to be interrupted, and assign it the role of RP-mapping agent.</p> <p>For scope ttl, specify the time-to-live value in hops to limit the RP discovery packets. All devices within the hop count from the source device receive the Auto-RP discovery messages. These messages tell other devices which group-to-RP mapping to use to avoid conflicts (such as overlapping group-to-RP ranges). There is no default setting. The range is 1 to 255.</p>
Step 7	<p>end</p> <p>Example:</p> <pre>Device(config)# end</pre>	Returns to privileged EXEC mode.
Step 8	<p>show running-config</p> <p>Example:</p> <pre>Device# show running-config</pre>	Verifies your entries.
Step 9	<p>show ip pim rp mapping</p> <p>Example:</p> <pre>Device# show ip pim rp mapping</pre>	Displays active RPs that are cached with associated multicast routing entries.
Step 10	<p>show ip pim rp</p> <p>Example:</p> <pre>Device# show ip pim rp</pre>	Displays the information cached in the routing table.
Step 11	<p>copy running-config startup-config</p> <p>Example:</p> <pre>Device# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Adding Auto-RP to an Existing Sparse-Mode Cloud

This section contains suggestions for the initial deployment of Auto-RP into an existing sparse-mode cloud to minimize disruption of the existing multicast infrastructure.

This procedure is optional.

SUMMARY STEPS

1. **enable**
2. **show running-config**
3. **configure terminal**
4. **ip pim send-rp-announce** *interface-id* **scope** *t1* **group-list** *access-list-number* **interval** *seconds*
5. **access-list** *access-list-number* {**deny** | **permit**} *source* [*source-wildcard*]
6. **ip pim send-rp-discovery** *scope* *t1*
7. **end**
8. **show running-config**
9. **show ip pim rp mapping**
10. **show ip pim rp**
11. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	show running-config Example: Device# show running-config	Verifies that a default RP is already configured on all PIM devices and the RP in the sparse-mode network. It was previously configured with the ip pim rp-address global configuration command. Note This step is not required for sparse-dense-mode environments. The selected RP should have good connectivity and be available across the network. Use this RP for the global groups (for example, 224.x.x.x and other global groups). Do not reconfigure the group address range that this RP serves. RPs dynamically discovered through Auto-RP take precedence over statically configured RPs. Assume that it is desirable to use a second RP for the local groups.
Step 3	configure terminal Example:	Enters global configuration mode.

	Command or Action	Purpose
	Device# <code>configure terminal</code>	
Step 4	<p>ip pim send-rp-announce <i>interface-id</i> scope <i>ttl</i> group-list <i>access-list-number</i> interval <i>seconds</i></p> <p>Example:</p> <pre>Device(config)# ip pim send-rp-announce gigabitethernet 1/0/5 scope 20 group-list 10 interval 120</pre>	<p>Configures another PIM device to be the candidate RP for local groups.</p> <ul style="list-style-type: none"> For <i>interface-id</i>, enter the interface type and number that identifies the RP address. Valid interfaces include physical ports, port channels, and VLANs. For scope <i>ttl</i>, specify the time-to-live value in hops. Enter a hop count that is high enough so that the RP-announce messages reach all mapping agents in the network. There is no default setting. The range is 1 to 255. For group-list <i>access-list-number</i>, enter an IP standard access list number from 1 to 99. If no access list is configured, the RP is used for all groups. For interval <i>seconds</i>, specify how often the announcement messages must be sent. The default is 60 seconds. The range is 1 to 16383. <p>Note To remove the PIM device configured as the candidate RP, use the no ip pim send-rp-announce <i>interface-id</i> global configuration command.</p>
Step 5	<p>access-list <i>access-list-number</i> {deny permit} <i>source</i> [<i>source-wildcard</i>]</p> <p>Example:</p> <pre>Device(config)# access-list 10 permit 224.0.0.0 15.255.255.255</pre>	<p>Creates a standard access list, repeating the command as many times as necessary.</p> <ul style="list-style-type: none"> For <i>access-list-number</i>, enter the access list number specified in Step 3. The deny keyword denies access if the conditions are matched. The permit keyword permits access if the conditions are matched. For <i>source</i>, enter the multicast group address range for which the RP should be used. (Optional) For <i>source-wildcard</i>, enter the wildcard bits in dotted decimal notation to be applied to the source. Place ones in the bit positions that you want to ignore. <p>Recall that the access list is always terminated by an implicit deny statement for everything.</p>

	Command or Action	Purpose
Step 6	<p>ip pim send-rp-discovery scope <i>t</i></p> <p>Example:</p> <pre>Device(config)# ip pim send-rp-discovery scope 50</pre>	<p>Finds a device whose connectivity is not likely to be interrupted, and assigns it the role of RP-mapping agent.</p> <p>For scope <i>t</i>, specify the time-to-live value in hops to limit the RP discovery packets. All devices within the hop count from the source device receive the Auto-RP discovery messages. These messages tell other devices which group-to-RP mapping to use to avoid conflicts (such as overlapping group-to-RP ranges). There is no default setting. The range is 1 to 255.</p> <p>Note To remove the device as the RP-mapping agent, use the no ip pim send-rp-discovery global configuration command.</p>
Step 7	<p>end</p> <p>Example:</p> <pre>Device(config)# end</pre>	Returns to privileged EXEC mode.
Step 8	<p>show running-config</p> <p>Example:</p> <pre>Device# show running-config</pre>	Verifies your entries.
Step 9	<p>show ip pim rp mapping</p> <p>Example:</p> <pre>Device# show ip pim rp mapping</pre>	Displays active RPs that are cached with associated multicast routing entries.
Step 10	<p>show ip pim rp</p> <p>Example:</p> <pre>Device# show ip pim rp</pre>	Displays the information cached in the routing table.
Step 11	<p>copy running-config startup-config</p> <p>Example:</p> <pre>Device# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Configuring Sparse Mode with a Single Static RP(CLI)

A rendezvous point (RP) is required in networks running Protocol Independent Multicast sparse mode (PIM-SM). In PIM-SM, traffic will be forwarded only to network segments with active receivers that have explicitly requested multicast data.

This section describes how to configure sparse mode with a single static RP.

Before you begin

All access lists that are needed when sparse mode is configured with a single static RP should be configured prior to beginning the configuration task.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip multicast-routing [distributed]**
4. **interface** *type number*
5. **ip pim sparse-mode**
6. Repeat Steps 1 through 5 on every interface that uses IP multicast.
7. **exit**
8. **ip pim rp-address** *rp-address* [*access-list*] [**override**]
9. **end**
10. **show ip pim rp** [**mapping**] [*rp-address*]
11. **show ip igmp groups** [*group-name* | *group-address*] [*interface-type interface-number*] [**detail**]
12. **show ip mroute**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: device> enable	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	configure terminal Example: device# configure terminal	Enters global configuration mode.
Step 3	ip multicast-routing [distributed] Example: device(config)# ip multicast-routing	Enables IP multicast routing. • Use the distributed keyword to enable Multicast Distributed Switching.
Step 4	interface <i>type number</i> Example:	Selects an interface that is connected to hosts on which PIM can be enabled.

	Command or Action	Purpose
	<code>device(config)# interface gigabitethernet 1/0/0</code>	
Step 5	ip pim sparse-mode Example: <code>device(config-if)# ip pim sparse-mode</code>	Enables PIM on an interface. You must use sparse mode.
Step 6	Repeat Steps 1 through 5 on every interface that uses IP multicast.	--
Step 7	exit Example: <code>device(config-if)# exit</code>	Returns to global configuration mode.
Step 8	ip pim rp-address <i>rp-address</i> [<i>access-list</i>] [override] Example: <code>device(config)# ip pim rp-address 192.168.0.0</code>	Configures the address of a PIM RP for a particular group. <ul style="list-style-type: none"> The optional <i>access-list</i> argument is used to specify the number or name a standard access list that defines the multicast groups to be statically mapped to the RP. <p>Note If no access list is defined, the RP will map to all multicast groups, 224/4.</p> <ul style="list-style-type: none"> The optional override keyword is used to specify that if dynamic and static group-to-RP mappings are used together and there is an RP address conflict, the RP address configured for a static group-to-RP mapping will take precedence. <p>Note If the override keyword is not specified and there is RP address conflict, dynamic group-to-RP mappings will take precedence over static group-to-RP mappings.</p>
Step 9	end Example: <code>device(config)# end</code>	Ends the current configuration session and returns to EXEC mode.
Step 10	show ip pim rp [mapping] [<i>rp-address</i>] Example: <code>device# show ip pim rp mapping</code>	(Optional) Displays RPs known in the network and shows how the router learned about each RP.
Step 11	show ip igmp groups [<i>group-name</i> <i>group-address</i> <i>interface-type interface-number</i>] [detail] Example:	(Optional) Displays the multicast groups having receivers that are directly connected to the router and that were learned through IGMP.

	Command or Action	Purpose
	device# <code>show ip igmp groups</code>	<ul style="list-style-type: none"> A receiver must be active on the network at the time that this command is issued in order for receiver information to be present on the resulting display.
Step 12	show ip mroute Example: device# <code>show ip mroute</code>	(Optional) Displays the contents of the IP mroute table.

Preventing Join Messages to False RPs

Determine whether the **ip pim accept-rp** command was previously configured throughout the network by using the **show running-config** privileged EXEC command. If the **ip pim accept-rp** command is not configured on any device, this problem can be addressed later. In those routers or multilayer devices already configured with the **ip pim accept-rp** command, you must enter the command again to accept the newly advertised RP.

To accept all RPs advertised with Auto-RP and reject all other RPs by default, use the **ip pim accept-rp auto-rp** global configuration command.

If all interfaces are in sparse mode, use a default-configured RP to support the two well-known groups 224.0.1.39 and 224.0.1.40. Auto-RP uses these two well-known groups to collect and distribute RP-mapping information. When this is the case and the **ip pim accept-rp auto-rp** command is configured, another **ip pim accept-rp** command accepting the RP must be configured as follows:

```
Switch(config)# ip pim accept-rp 172.10.20.1 1
Switch(config)# access-list 1 permit 224.0.1.39
Switch(config)# access-list 1 permit 224.0.1.40
```

This procedure is optional.

Filtering Incoming RP Announcement Messages

You can add configuration commands to the mapping agents to prevent a maliciously configured router from masquerading as a candidate RP and causing problems.

This procedure is optional.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip pim rp-announce-filter rp-list *access-list-number* group-list *access-list-number***
4. **access-list *access-list-number* {deny | permit} source [*source-wildcard*]**
5. **end**
6. **show running-config**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	ip pim rp-announce-filter rp-list access-list-number group-list access-list-number Example: Device(config)# ip pim rp-announce-filter rp-list 10 group-list 14	Filters incoming RP announcement messages. Enter this command on each mapping agent in the network. Without this command, all incoming RP-announce messages are accepted by default. For rp-list access-list-number , configure an access list of candidate RP addresses that, if permitted, is accepted for the group ranges supplied in the group-list access-list-number variable. If this variable is omitted, the filter applies to all multicast groups. If more than one mapping agent is used, the filters must be consistent across all mapping agents to ensure that no conflicts occur in the group-to-RP mapping information. Note To remove a filter on incoming RP announcement messages, use the no ip pim rp-announce-filter rp-list access-list-number [group-list access-list-number] global configuration command.
Step 4	access-list access-list-number {deny permit} source [source-wildcard] Example: Device(config)# access-list 10 permit 10.8.1.0 255.255.224.0	Creates a standard access list, repeating the command as many times as necessary. <ul style="list-style-type: none"> • For <i>access-list-number</i>, enter the access list number specified in Step 2. • The deny keyword denies access if the conditions are matched. • The permit keyword permits access if the conditions are matched. • Create an access list that specifies from which routers and multilayer devices the mapping agent accepts candidate RP announcements (rp-list ACL).

	Command or Action	Purpose
		<ul style="list-style-type: none"> • Create an access list that specifies the range of multicast groups from which to accept or deny (group-list ACL). • For <i>source</i>, enter the multicast group address range for which the RP should be used. • (Optional) For <i>source-wildcard</i>, enter the wildcard bits in dotted decimal notation to be applied to the source. Place ones in the bit positions that you want to ignore. <p>The access list is always terminated by an implicit deny statement for everything.</p>
Step 5	end Example: Device(config)# end	Returns to privileged EXEC mode.
Step 6	show running-config Example: Device# show running-config	Verifies your entries.
Step 7	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Configuring PIMv2 BSR

The process for configuring PIMv2 BSR may involve the following optional tasks:

- Defining the PIM domain border
- Defining the IP multicast boundary
- Configuring candidate BSRs
- Configuring candidate RPs

Defining the PIM Domain Border

Perform the following steps to configure the PIM domain border. This procedure is optional.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface *interface-id***
4. **ip pim bsr-border**
5. **end**
6. **show running-config**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Device> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: <pre>Device# configure terminal</pre>	Enters global configuration mode.
Step 3	interface <i>interface-id</i> Example: <pre>Device(config)# interface gigabitethernet 1/0/1</pre>	Specifies the interface to be configured, and enters interface configuration mode.
Step 4	ip pim bsr-border Example: <pre>Device(config-if)# ip pim bsr-border</pre>	Defines a PIM bootstrap message boundary for the PIM domain. Enter this command on each interface that connects to other bordering PIM domains. This command instructs the device to neither send nor receive PIMv2 BSR messages on this interface. Note To remove the PIM border, use the no ip pim bsr-border interface configuration command.
Step 5	end Example: <pre>Device(config)# end</pre>	Returns to privileged EXEC mode.
Step 6	show running-config Example:	Verifies your entries.

	Command or Action	Purpose
	Device# <code>show running-config</code>	
Step 7	copy running-config startup-config Example: Device# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Defining the IP Multicast Boundary

You define a multicast boundary to prevent Auto-RP messages from entering the PIM domain. You create an access list to deny packets destined for 224.0.1.39 and 224.0.1.40, which carry Auto-RP information.

This procedure is optional.

SUMMARY STEPS

1. `enable`
2. `configure terminal`
3. `access-list access-list-number deny source [source-wildcard]`
4. `interface interface-id`
5. `ip multicast boundary access-list-number`
6. `end`
7. `show running-config`
8. `copy running-config startup-config`

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> <code>enable</code>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode.
Step 3	access-list access-list-number deny source [source-wildcard] Example: Device(config)# <code>access-list 12 deny 224.0.1.39</code>	Creates a standard access list, repeating the command as many times as necessary. <ul style="list-style-type: none"> • For <i>access-list-number</i>, the range is 1 to 99. • The deny keyword denies access if the conditions are matched.

	Command or Action	Purpose
	<code>access-list 12 deny 224.0.1.40</code>	<ul style="list-style-type: none"> For <i>source</i>, enter multicast addresses 224.0.1.39 and 224.0.1.40, which carry Auto-RP information. (Optional) For <i>source-wildcard</i>, enter the wildcard bits in dotted decimal notation to be applied to the source. Place ones in the bit positions that you want to ignore. <p>The access list is always terminated by an implicit deny statement for everything.</p>
Step 4	interface <i>interface-id</i> Example: <pre>Device(config)# interface gigabitethernet 1/0/1</pre>	Specifies the interface to be configured, and enters interface configuration mode.
Step 5	ip multicast boundary <i>access-list-number</i> Example: <pre>Device(config-if)# ip multicast boundary 12</pre>	Configures the boundary, specifying the access list you created in Step 2. Note To remove the boundary, use the no ip multicast boundary interface configuration command.
Step 6	end Example: <pre>Device(config)# end</pre>	Returns to privileged EXEC mode.
Step 7	show running-config Example: <pre>Device# show running-config</pre>	Verifies your entries.
Step 8	copy running-config startup-config Example: <pre>Device# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Configuring Candidate BSRs

You can configure one or more candidate BSRs. The devices serving as candidate BSRs should have good connectivity to other devices and be in the backbone portion of the network.

This procedure is optional.

SUMMARY STEPS

1. enable
2. configure terminal
3. ip pim bsr-candidate *interface-id hash-mask-length* [*priority*]
4. end
5. show running-config
6. copy running-config startup-config

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p>enable</p> <p>Example:</p> <pre>Device> enable</pre>	<p>Enables privileged EXEC mode.</p> <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	<p>configure terminal</p> <p>Example:</p> <pre>Device# configure terminal</pre>	<p>Enters global configuration mode.</p>
Step 3	<p>ip pim bsr-candidate <i>interface-id hash-mask-length</i> [<i>priority</i>]</p> <p>Example:</p> <pre>Device(config)# ip pim bsr-candidate gigabitethernet 1/0/3 28 100</pre>	<p>Configures your device to be a candidate BSR.</p> <ul style="list-style-type: none"> • For <i>interface-id</i>, enter the interface on this device from which the BSR address is derived to make it a candidate. This interface must be enabled with PIM. Valid interfaces include physical ports, port channels, and VLANs. • For <i>hash-mask-length</i>, specify the mask length (32 bits maximum) that is to be ANDed with the group address before the hash function is called. All groups with the same seed hash correspond to the same RP. For example, if this value is 24, only the first 24 bits of the group addresses matter. • (Optional) For <i>priority</i>, enter a number from 0 to 255. The BSR with the larger priority is preferred. If the priority values are the same, the device with the highest IP address is selected as the BSR. The default is 0. <p>Note To remove this device as a candidate BSR, use the no ip pim bsr-candidate global configuration command.</p>
Step 4	<p>end</p> <p>Example:</p>	<p>Returns to privileged EXEC mode.</p>

	Command or Action	Purpose
	Device(config)# end	
Step 5	show running-config Example: Device# show running-config	Verifies your entries.
Step 6	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Configuring the Candidate RPs

You can configure one or more candidate RPs. Similar to BSRs, the RPs should also have good connectivity to other devices and be in the backbone portion of the network. An RP can serve the entire IP multicast address space or a portion of it. Candidate RPs send candidate RP advertisements to the BSR.

This procedure is optional.

Before you begin

When deciding which devices should be RPs, consider these options:

- In a network of Cisco routers and multilayer devices where only Auto-RP is used, any device can be configured as an RP.
- In a network that includes only Cisco PIMv2 routers and multilayer devices and with routers from other vendors, any device can be used as an RP.
- In a network of Cisco PIMv1 routers, Cisco PIMv2 routers, and routers from other vendors, configure only Cisco PIMv2 routers and multilayer devices as RPs.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip pim rp-candidate** *interface-id* [**group-list** *access-list-number*]
4. **access-list** *access-list-number* {**deny** | **permit**} *source* [*source-wildcard*]
5. **end**
6. **show running-config**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p>enable</p> <p>Example:</p> <pre>Device> enable</pre>	<p>Enables privileged EXEC mode.</p> <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	<p>configure terminal</p> <p>Example:</p> <pre>Device# configure terminal</pre>	<p>Enters global configuration mode.</p>
Step 3	<p>ip pim rp-candidate <i>interface-id</i> [group-list <i>access-list-number</i>]</p> <p>Example:</p> <pre>Device(config)# ip pim rp-candidate gigabitethernet 1/0/5 group-list 10</pre>	<p>Configures your device to be a candidate RP.</p> <ul style="list-style-type: none"> • For <i>interface-id</i>, specify the interface whose associated IP address is advertised as a candidate RP address. Valid interfaces include physical ports, port channels, and VLANs. • (Optional) For group-list <i>access-list-number</i>, enter an IP standard access list number from 1 to 99. If no group-list is specified, the device is a candidate RP for all groups. <p>Note To remove this device as a candidate RP, use the no ip pim rp-candidate <i>interface-id</i> global configuration command.</p>
Step 4	<p>access-list <i>access-list-number</i> {deny permit} <i>source</i> [<i>source-wildcard</i>]</p> <p>Example:</p> <pre>Device(config)# access-list 10 permit 239.0.0.0 0.255.255.255</pre>	<p>Creates a standard access list, repeating the command as many times as necessary.</p> <ul style="list-style-type: none"> • For <i>access-list-number</i>, enter the access list number specified in Step 2. • The deny keyword denies access if the conditions are matched. The permit keyword permits access if the conditions are matched. • For <i>source</i>, enter the number of the network or host from which the packet is being sent. • (Optional) For <i>source-wildcard</i>, enter the wildcard bits in dotted decimal notation to be applied to the source. Place ones in the bit positions that you want to ignore. <p>The access list is always terminated by an implicit deny statement for everything.</p>

	Command or Action	Purpose
Step 5	end Example: Device(config)# end	Returns to privileged EXEC mode.
Step 6	show running-config Example: Device# show running-config	Verifies your entries.
Step 7	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Delaying the Use of PIM Shortest-Path Tree

Perform these steps to configure a traffic rate threshold that must be reached before multicast routing is switched from the source tree to the shortest-path tree.

This procedure is optional.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **access-list *access-list-number* {deny | permit} source [*source-wildcard*]**
4. **ip pim spt-threshold {*kpbs* | infinity} [*group-list access-list-number*]**
5. **end**
6. **show running-config**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example:	Enters global configuration mode.

	Command or Action	Purpose
	Device# <code>configure terminal</code>	
Step 3	<p>access-list <i>access-list-number</i> {deny permit} <i>source</i> [<i>source-wildcard</i>]</p> <p>Example:</p> <pre>Device(config)# access-list 16 permit 225.0.0.0 0.255.255.255</pre>	<p>Creates a standard access list.</p> <ul style="list-style-type: none"> For <i>access-list-number</i>, the range is 1 to 99. The deny keyword denies access if the conditions are matched. The permit keyword permits access if the conditions are matched. For <i>source</i>, specify the multicast group to which the threshold will apply. (Optional) For <i>source-wildcard</i>, enter the wildcard bits in dotted decimal notation to be applied to the source. Place ones in the bit positions that you want to ignore. <p>The access list is always terminated by an implicit deny statement for everything.</p>
Step 4	<p>ip pim spt-threshold {<i>kbps</i> infinity} [group-list <i>access-list-number</i>]</p> <p>Example:</p> <pre>Device(config)# ip pim spt-threshold infinity group-list 16</pre>	<p>Specifies the threshold that must be reached before moving to shortest-path tree (spt).</p> <ul style="list-style-type: none"> For <i>kbps</i>, specify the traffic rate in kilobits per second. The default is 0 kbps. <p>Note Because of device hardware limitations, 0 kbps is the only valid entry even though the range is 0 to 4294967.</p> <ul style="list-style-type: none"> Specify infinity if you want all sources for the specified group to use the shared tree, never switching to the source tree. (Optional) For group-list <i>access-list-number</i>, specify the access list created in Step 2. If the value is 0 or if the group list is not used, the threshold applies to all groups. <p>Note To return to the default setting, use the no ip pim spt-threshold {kbps infinity} global configuration command.</p>
Step 5	<p>end</p> <p>Example:</p> <pre>Device(config)# end</pre>	<p>Returns to privileged EXEC mode.</p>

	Command or Action	Purpose
Step 6	show running-config Example: Device# <code>show running-config</code>	Verifies your entries.
Step 7	copy running-config startup-config Example: Device# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Modifying the PIM Router-Query Message Interval

PIM routers and multilayer devices send PIM router-query messages to find which device will be the designated router (DR) for each LAN segment (subnet). The DR is responsible for sending IGMP host-query messages to all hosts on the directly connected LAN.

With PIM DM operation, the DR has meaning only if IGMPv1 is in use. IGMPv1 does not have an IGMP querier election process, so the elected DR functions as the IGMP querier. With PIM-SM operation, the DR is the device that is directly connected to the multicast source. It sends PIM register messages to notify the RP that multicast traffic from a source needs to be forwarded down the shared tree. In this case, the DR is the device with the highest IP address.

This procedure is optional.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **ip pim query-interval** *seconds*
5. **end**
6. **show ip igmp interface** [*interface-id*]
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> <code>enable</code>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example:	Enters global configuration mode.

	Command or Action	Purpose
	Device# <code>configure terminal</code>	
Step 3	interface <i>interface-id</i> Example: Device(config)# interface gigabitethernet 1/0/1	Specifies the interface to be configured, and enters interface configuration mode.
Step 4	ip pim query-interval <i>seconds</i> Example: Device(config-if)# ip pim query-interval 45	Configures the frequency at which the device sends PIM router-query messages. The default is 30 seconds. The range is 1 to 65535. Note To return to the default setting, use the no ip pim query-interval [seconds] interface configuration command.
Step 5	end Example: Device(config)# end	Returns to privileged EXEC mode.
Step 6	show ip igmp interface [<i>interface-id</i>] Example: Device# show ip igmp interface	Verifies your entries.
Step 7	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Verifying PIM Operations

Verifying IP Multicast Operation in a PIM-SM or a PIM-SSM Network

When you verify the operation of IP multicast in a PIM-SM network environment or in a PIM-SSM network environment, a useful approach is to begin the verification process on the last hop router, and then continue the verification process on the routers along the SPT until the first hop router has been reached. The goal of the verification is to ensure that IP multicast traffic is being routed properly through an IP multicast network.

Perform the following optional tasks to verify IP multicast operation in a PIM-SM or a PIM-SSM network. The steps in these tasks help to locate a faulty hop when sources and receivers are not operating as expected.



Note If packets are not reaching their expected destinations, you might want consider disabling IP multicast fast switching, which would place the router in process switching mode. If packets begin reaching their proper destinations after IP multicast fast switching has been disabled, then the issue most likely was related to IP multicast fast switching.

Verifying IP Multicast on the First Hop Router

Enter these commands on the first hop router to verify IP multicast operations on the first hop router:

SUMMARY STEPS

1. **enable**
2. **show ip mroute** [*group-address*]
3. **show ip mroute active** [*kb/s*]

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	show ip mroute [<i>group-address</i>] Example: Device# show ip mroute 239.1.2.3 (*, 239.1.2.3), 00:18:10/stopped, RP 172.16.0.1, flags: SPF Incoming interface: Serial1/0, RPF nbr 172.31.200.2 Outgoing interface list: Null (10.0.0.1, 239.1.2.3), 00:18:10/00:03:22, flags: FT Incoming interface: GigabitEthernet0/0/0, RPF nbr 0.0.0.0 Outgoing interface list: Serial1/0, Forward/Sparse-Dense, 00:18:10/00:03:19	Confirms that the F flag has been set for mroutes on the first hop router.
Step 3	show ip mroute active [<i>kb/s</i>] Example: Device# show ip mroute active Active IP Multicast Sources - sending >= 4 kbps	Displays information about active multicast sources sending to groups. The output of this command provides information about the multicast packet rate for active sources.

	Command or Action	Purpose
	<pre>Group: 239.1.2.3, (?) Source: 10.0.0.1 (?) Rate: 20 pps/4 kbps(1sec), 4 kbps(last 30 secs), 4 kbps(life avg)</pre>	<p>Note By default, the output of the show ip mroute command with the active keyword displays information about active sources sending traffic to groups at a rate greater than or equal to 4 kb/s. To display information about active sources sending low-rate traffic to groups (that is, traffic less than 4 kb/s), specify a value of 1 for the <i>kb/s</i> argument. Specifying a value of 1 for this argument displays information about active sources sending traffic to groups at a rate equal to or greater than 1 kb/s, which effectively displays information about all possible active source traffic.</p>

Verifying IP Multicast on Routers Along the SPT

Enter these commands on routers along the SPT to verify IP multicast operations on routers along the SPT in a PIM-SM or PIM-SSM network:

SUMMARY STEPS

1. **enable**
2. **show ip mroute** [*group-address*]
3. **show ip mroute active**

DETAILED STEPS

	Command or Action	Purpose
<p>Step 1</p>	<p>enable</p> <p>Example:</p> <pre>Device> enable</pre>	<p>Enables privileged EXEC mode.</p> <ul style="list-style-type: none"> • Enter your password if prompted.
<p>Step 2</p>	<p>show ip mroute [<i>group-address</i>]</p> <p>Example:</p> <pre>Device# show ip mroute 239.1.2.3 (*, 239.1.2.3), 00:17:56/00:03:02, RP 172.16.0.1, flags: S Incoming interface: Null, RPF nbr 0.0.0.0 Outgoing interface list: GigabitEthernet0/0/0, Forward/Sparse-Dense, 00:17:56/00:03:02 (10.0.0.1, 239.1.2.3), 00:15:34/00:03:28, flags: T Incoming interface: Serial1/0, RPF nbr 172.31.200.1 Outgoing interface list: GigabitEthernet0/0/0, Forward/Sparse-Dense, 00:15:34/00:03:02</pre>	<p>Confirms the RPF neighbor towards the source for a particular group or groups.</p>

	Command or Action	Purpose
Step 3	<p>show ip mroute active</p> <p>Example:</p> <pre>Device# show ip mroute active Active IP Multicast Sources - sending >= 4 kbps Group: 239.1.2.3, (?) Source: 10.0.0.1 (?) Rate: 20 pps/4 kbps(1sec), 4 kbps(last 30 secs), 4 kbps(life avg)</pre>	<p>Displays information about active multicast sources sending to groups. The output of this command provides information about the multicast packet rate for active sources.</p> <p>Note By default, the output of the show ip mroute command with the active keyword displays information about active sources sending traffic to groups at a rate greater than or equal to 4 kb/s. To display information about active sources sending low-rate traffic to groups (that is, traffic less than 4 kb/s), specify a value of 1 for the <i>kb/s</i> argument. Specifying a value of 1 for this argument displays information about active sources sending traffic to groups at a rate equal to or greater than 1 kb/s, which effectively displays information about all possible active source traffic.</p>

Verifying IP Multicast Operation on the Last Hop Router

Enter these commands on the last hop router to verify IP multicast operations on the last hop router:

SUMMARY STEPS

1. **enable**
2. **show ip igmp groups**
3. **show ip pim rp mapping**
4. **show ip mroute**
5. **show ip interface** [*type number*]
6. **show ip pim interface count**
7. **show ip mroute count**
8. **show ip mroute active** [*kb/s*]

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p>enable</p> <p>Example:</p> <pre>Device> enable</pre>	<p>Enables privileged EXEC mode.</p> <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	<p>show ip igmp groups</p> <p>Example:</p> <pre>Device# show ip igmp groups IGMP Connected Group Membership Group Address Interface Uptime Expires Last Reporter</pre>	<p>Verifies IGMP memberships on the last hop router. This information will confirm the multicast groups with receivers that are directly connected to the last hop router and that are learned through IGMP.</p>

	Command or Action	Purpose
	<pre>239.1.2.3 GigabitEthernet1/0/0 00:05:14 00:02:14 10.1.0.6 224.0.1.39 GigabitEthernet0/0/0 00:09:11 00:02:08 172.31.100.1</pre>	
Step 3	<p>show ip pim rp mapping</p> <p>Example:</p> <pre>Device# show ip pim rp mapping PIM Group-to-RP Mappings Group(s) 224.0.0.0/4 RP 172.16.0.1 (?), v2v1 Info source: 172.16.0.1 (?), elected via Auto-RP Uptime: 00:09:11, expires: 00:02:47</pre>	<p>Confirms that the group-to-RP mappings are being populated correctly on the last hop router.</p> <p>Note Ignore this step if you are verifying a last hop router in a PIM-SSM network. The show ip pim rp mapping command does not work with routers in a PIM-SSM network because PIM-SSM does not use RPs. In addition, if configured correctly, PIM-SSM groups do not appear in the output of the show ip pim rp mapping command.</p>
Step 4	<p>show ip mroute</p> <p>Example:</p> <pre>Device# show ip mroute (*, 239.1.2.3), 00:05:14/00:03:04, RP 172.16.0.1, flags: SJC Incoming interface: GigabitEthernet0/0/0, RPF nbr 172.31.100.1 Outgoing interface list: GigabitEthernet1/0, Forward/Sparse-Dense, 00:05:10/00:03:04 (10.0.0.1, 239.1.2.3), 00:02:49/00:03:29, flags: T Incoming interface: GigabitEthernet0/0/0, RPF nbr 172.31.100.1 Outgoing interface list: GigabitEthernet1/0, Forward/Sparse-Dense, 00:02:49/00:03:04 (*, 224.0.1.39), 00:10:05/stopped, RP 0.0.0.0, flags: DC Incoming interface: Null, RPF nbr 0.0.0.0 Outgoing interface list: GigabitEthernet1/0, Forward/Sparse-Dense, 00:05:15/00:00:00 GigabitEthernet0/0, Forward/Sparse-Dense, 00:10:05/00:00:00 (172.16.0.1, 224.0.1.39), 00:02:00/00:01:33, flags: PTX Incoming interface: GigabitEthernet0/0/0, RPF nbr 172.31.100.1</pre>	<p>Verifies that the mroute table is being populated properly on the last hop router.</p>
Step 5	<p>show ip interface [type number]</p> <p>Example:</p>	<p>Verifies that multicast fast switching is enabled for optimal performance on the outgoing interface on the last hop router.</p>

	Command or Action	Purpose																		
	<pre>Device# show ip interface GigabitEthernet 0/0/0 GigabitEthernet0/0 is up, line protocol is up Internet address is 172.31.100.2/24 Broadcast address is 255.255.255.255 Address determined by setup command MTU is 1500 bytes Helper address is not set Directed broadcast forwarding is disabled Multicast reserved groups joined: 224.0.0.1 224.0.0.22 224.0.0.13 224.0.0.5 224.0.0.6 Outgoing access list is not set Inbound access list is not set Proxy ARP is enabled Local Proxy ARP is disabled Security level is default Split horizon is enabled ICMP redirects are always sent ICMP unreachable are always sent ICMP mask replies are never sent IP fast switching is enabled IP fast switching on the same interface is disabled IP Flow switching is disabled IP CEF switching is disabled IP Fast switching turbo vector IP multicast fast switching is enabled IP multicast distributed fast switching is disabled IP route-cache flags are Fast Router Discovery is disabled IP output packet accounting is disabled IP access violation accounting is disabled TCP/IP header compression is disabled RTP/IP header compression is disabled Policy routing is disabled Network address translation is disabled WCCP Redirect outbound is disabled WCCP Redirect inbound is disabled WCCP Redirect exclude is disabled BGP Policy Mapping is disabled</pre>	<p>Note Using the no ip mroute-cache interface command disables IP multicast fast-switching. When IP multicast fast switching is disabled, packets are forwarded through the process-switched path.</p>																		
<p>Step 6</p>	<p>show ip pim interface count</p> <p>Example:</p> <pre>Device# show ip pim interface count</pre> <p>State: * - Fast Switched, D - Distributed Fast Switched H - Hardware Switching Enabled</p> <table border="1"> <thead> <tr> <th>Address</th> <th>Interface</th> <th>FS</th> </tr> </thead> <tbody> <tr> <td>Mpackets In/Out</td> <td></td> <td></td> </tr> <tr> <td>172.31.100.2</td> <td>GigabitEthernet0/0/0</td> <td>*</td> </tr> <tr> <td>4122/0</td> <td></td> <td></td> </tr> <tr> <td>10.1.0.1</td> <td>GigabitEthernet1/0/0</td> <td>*</td> </tr> <tr> <td>0/3193</td> <td></td> <td></td> </tr> </tbody> </table>	Address	Interface	FS	Mpackets In/Out			172.31.100.2	GigabitEthernet0/0/0	*	4122/0			10.1.0.1	GigabitEthernet1/0/0	*	0/3193			<p>Confirms that multicast traffic is being forwarded on the last hop router.</p>
Address	Interface	FS																		
Mpackets In/Out																				
172.31.100.2	GigabitEthernet0/0/0	*																		
4122/0																				
10.1.0.1	GigabitEthernet1/0/0	*																		
0/3193																				
<p>Step 7</p>	<p>show ip mroute count</p> <p>Example:</p>	<p>Confirms that multicast traffic is being forwarded on the last hop router.</p>																		

	Command or Action	Purpose
	<pre>Device# show ip mroute count IP Multicast Statistics 6 routes using 4008 bytes of memory 3 groups, 1.00 average sources per group Forwarding Counts: Pkt Count/Pkts per second/Avg Pkt Size/Kilobits per second Other counts: Total/RPF failed/Other drops(OIF-null, rate-limit etc) Group: 239.1.2.3, Source count: 1, Packets forwarded: 3165, Packets received: 3165 RP-tree: Forwarding: 0/0/0/0, Other: 0/0/0 Source: 10.0.0.1/32, Forwarding: 3165/20/28/4, Other: 0/0/0 Group: 224.0.1.39, Source count: 1, Packets forwarded: 21, Packets received: 120 Source: 172.16.0.1/32, Forwarding: 21/1/48/0, Other: 120/0/99 Group: 224.0.1.40, Source count: 1, Packets forwarded: 10, Packets received: 10 Source: 172.16.0.1/32, Forwarding: 10/1/48/0, Other: 10/0/0</pre>	
Step 8	<p>show ip mroute active [<i>kb/s</i>]</p> <p>Example:</p> <pre>Device# show ip mroute active Active IP Multicast Sources - sending >= 4 kbps Group: 239.1.2.3, (?) Source: 10.0.0.1 (?) Rate: 20 pps/4 kbps(1sec), 4 kbps(last 50 secs), 4 kbps(life avg)</pre>	<p>Displays information about active multicast sources sending traffic to groups on the last hop router. The output of this command provides information about the multicast packet rate for active sources.</p> <p>Note By default, the output of the show ip mroute command with the active keyword displays information about active sources sending traffic to groups at a rate greater than or equal to 4 kb/s. To display information about active sources sending low-rate traffic to groups (that is, traffic less than 4 kb/s), specify a value of 1 for the <i>kb/s</i> argument. Specifying a value of 1 for this argument displays information about active sources sending traffic to groups at a rate equal to or greater than 1 kb/s, which effectively displays information about all possible active source traffic.</p>

Using PIM-Enabled Routers to Test IP Multicast Reachability

If all the PIM-enabled routers and access servers that you administer are members of a multicast group, pinging that group causes all routers to respond, which can be a useful administrative and debugging tool.

To use PIM-enabled routers to test IP multicast reachability, perform the following tasks:

Configuring Routers to Respond to Multicast Pings

Follow these steps to configure a router to respond to multicast pings. Perform the task on all the interfaces of a router and on all the routers participating in the multicast network:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *type number*
4. **ip igmp join-group** *group-address*
5. Repeat Step 3 and Step 4 for each interface on the router participating in the multicast network.
6. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	configure terminal	Enters global configuration mode.
Step 3	interface <i>type number</i>	Enters interface configuration mode. For the <i>type</i> and <i>number</i> arguments, specify an interface that is directly connected to hosts or is facing hosts.
Step 4	ip igmp join-group <i>group-address</i>	(Optional) Configures an interface on the router to join the specified group. For the purpose of this task, configure the same group address for the <i>group-address</i> argument on all interfaces on the router participating in the multicast network. Note With this method, the router accepts the multicast packets in addition to forwarding them. Accepting the multicast packets prevents the router from fast switching.
Step 5	Repeat Step 3 and Step 4 for each interface on the router participating in the multicast network.	--
Step 6	end	Ends the current configuration session and returns to privileged EXEC mode.

Pinging Routers Configured to Respond to Multicast Pings

Follow these steps on a router to initiate a ping test to the routers configured to respond to multicast pings. This task is used to test IP multicast reachability in a network.

SUMMARY STEPS

1. **enable**
2. **ping** *group-address*

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	ping <i>group-address</i>	Pings an IP multicast group address. A successful response indicates that the group address is functioning.

Monitoring and Troubleshooting PIM

Monitoring PIM Information

Use the privileged EXEC commands in the following table to monitor your PIM configurations.

Table 30: PIM Monitoring Commands

Command	Purpose
show ip pim all-vrfs tunnel [tunnel <i>tunnel_number</i> verbose]	Displays all VRFs.
show ip pim autorp	Displays global auto-RP information.
show ip pim boundary	Displays information about mroutes filtered by administratively scoped IPv4 multicast boundaries configured on an interface.
show ip pim interface	Displays information about interfaces configured for Protocol Independent Multicast (PIM).
show ip pim mdt [bgp]	Displays details about the Border Gateway Protocol (BGP) advertisement of the route distinguisher (RD) for the multicast distribution tree (MDT) default group.
show ip pim neighbor	Displays the PIM neighbor information.
show ip pim rp[<i>group-name</i> <i>group-address</i>]	Displays RP routers associated with a sparse-mode multicast group. This command is available in all software images.
show ip pim tunnel [tunnel verbose]	Displays information about Protocol Independent Multicast (PIM) tunnel interfaces
show ip pim vrf { word { all-vrfs autorp boundary bsr-router interface mdt neighbor rp rp-hash tunnel } }	Displays the VPN routing/forwarding instance.

Command	Purpose
<code>show ip igmp groups detail</code>	Displays the interested clients that have joined the specific multicast source group.
<code>show ip igmp snooping mroute</code>	Verifies that the multicast stream forwards from the source to the interested clients.

Monitoring the RP Mapping and BSR Information

Use the privileged EXEC mode in the following table to verify the consistency of group-to-RP mappings:

Table 31: RP Mapping Monitoring Commands

Command	Purpose
<code>show ip pim rp [hostname or IP address mapping [hostname or IP address elected in-use] metric [hostname or IP address]]</code>	<p>Displays all available RP mappings and metrics. This tells you how the device learns of the RP (through the BSR or the Auto-RP mechanism).</p> <ul style="list-style-type: none"> • (Optional) For the <i>hostname</i>, specify the IP name of the group about which to display RPs. • (Optional) For the <i>IP address</i>, specify the IP address of the group about which to display RPs. • (Optional) Use the mapping keyword to display all group-to-RP mappings of which the Cisco device is aware (either configured or learned from Auto-RP). • (Optional) Use the metric keyword to display the RP RPF metric.
<code>show ip pim rp-hash group</code>	Displays the RP that was selected for the specified group. That is, on a PIMv2 router or multilayer device, confirms that the same RP is the one that a PIMv1 system chooses. For <i>group</i> , enter the group address for which to display RP information.

Use the privileged EXEC commands in the following table to monitor BSR information:

Table 32: BSR Monitoring Commands

Command	Purpose
<code>show ip pim bsr</code>	Displays information about the elected BSR.
<code>show ip pim bsr-router</code>	Displays information about the BSRv2.

Troubleshooting PIMv1 and PIMv2 Interoperability Problems

When debugging interoperability problems between PIMv1 and PIMv2, check these in the order shown:

1. Verify RP mapping with the **show ip pim rp-hash** privileged EXEC command, making sure that all systems agree on the same RP for the same group.
2. Verify interoperability between different versions of DRs and RPs. Make sure that the RPs are interacting with the DRs properly (by responding with register-stops and forwarding decapsulated data packets from registers).

Configuration Examples for PIM

Example: Enabling PIM Stub Routing

In this example, IP multicast routing is enabled, Switch A PIM uplink port 25 is configured as a routed uplink port with **spare-dense-mode** enabled. PIM stub routing is enabled on the VLAN 100 interfaces and on Gigabit Ethernet port 20.

```
Device(config)# ip multicast-routing distributed
Device(config)# interface GigabitEthernet3/0/25
Device(config-if)# no switchport
Device(config-if)# ip address 3.1.1.2 255.255.255.0
Device(config-if)# ip pim sparse-dense-mode
Device(config-if)# exit
Device(config)# interface vlan100
Device(config-if)# ip pim passive
Device(config-if)# exit
Device(config)# interface GigabitEthernet3/0/20
Device(config-if)# ip pim passive
Device(config-if)# exit
Device(config)# interface vlan100
Device(config-if)# ip address 100.1.1.1 255.255.255.0
Device(config-if)# ip pim passive
Device(config-if)# exit
Device(config)# interface GigabitEthernet3/0/20
Device(config-if)# no switchport
Device(config-if)# ip address 10.1.1.1 255.255.255.0
Device(config-if)# ip pim passive
Device(config-if)# end
```

Example: Verifying PIM Stub Routing

To verify that PIM stub is enabled for each interface, use the **show ip pim interface** privileged EXEC command:

```
Device# show ip pim interface
Address Interface Ver/ Nbr Query DR DR
Mode Count Intvl Prior
3.1.1.2 GigabitEthernet3/0/25 v2/SD 1 30 1 3.1.1.2
100.1.1.1 Vlan100 v2/P 0 30 1 100.1.1.1
10.1.1.1 GigabitEthernet3/0/20 v2/P 0 30 1 10.1.1.1
```

Example: Manually Assigning an RP to Multicast Groups

This example shows how to configure the address of the RP to 147.106.6.22 for multicast group 225.2.2.2 only:

```
Device(config)# access-list 1 permit 225.2.2.2 0.0.0.0
Device(config)# ip pim rp-address 147.106.6.22 1
```

Example: Configuring Auto-RP

This example shows how to send RP announcements out all PIM-enabled interfaces for a maximum of 31 hops. The IP address of port 1 is the RP. Access list 5 describes the group for which this device serves as RP:

```
Device(config)# ip pim send-rp-announce gigabitethernet1/0/1 scope 31 group-list 5
Device(config)# access-list 5 permit 224.0.0.0 15.255.255.255
```

Example: Defining the IP Multicast Boundary to Deny Auto-RP Information

This example shows a portion of an IP multicast boundary configuration that denies Auto-RP information:

```
Device(config)# access-list 1 deny 224.0.1.39
Device(config)# access-list 1 deny 224.0.1.40
Device(config)# access-list 1 permit all
Device(config)# interface gigabitethernet1/0/1
Device(config-if)# ip multicast boundary 1
```

Example: Filtering Incoming RP Announcement Messages

This example shows a sample configuration on an Auto-RP mapping agent that is used to prevent candidate RP announcements from being accepted from unauthorized candidate RPs:

```
Device(config)# ip pim rp-announce-filter rp-list 10 group-list 20
Device(config)# access-list 10 permit host 172.16.5.1
Device(config)# access-list 10 permit host 172.16.2.1
Device(config)# access-list 20 deny 239.0.0.0 0.0.255.255
Device(config)# access-list 20 permit 224.0.0.0 15.255.255.255
```

The mapping agent accepts candidate RP announcements from only two devices, 172.16.5.1 and 172.16.2.1. The mapping agent accepts candidate RP announcements from these two devices only for multicast groups that fall in the group range of 224.0.0.0 to 239.255.255.255. The mapping agent does not accept candidate RP announcements from any other devices in the network. Furthermore, the mapping agent does not accept candidate RP announcements from 172.16.5.1 or 172.16.2.1 if the announcements are for any groups in the 239.0.0.0 through 239.255.255.255 range. This range is the administratively scoped address range.

Example: Preventing Join Messages to False RPs

If all interfaces are in sparse mode, use a default-configured RP to support the two well-known groups 224.0.1.39 and 224.0.1.40. Auto-RP uses these two well-known groups to collect and distribute RP-mapping information. When this is the case and the **ip pim accept-rp auto-rp** command is configured, another **ip pim accept-rp** command accepting the RP must be configured as follows:

```
Device(config)# ip pim accept-rp 172.10.20.1 1
Device(config)# access-list 1 permit 224.0.1.39
Device(config)# access-list 1 permit 224.0.1.40
```

Example: Configuring Candidate BSRs

This example shows how to configure a candidate BSR, which uses the IP address 172.21.24.18 on a port as the advertised BSR address, uses 30 bits as the hash-mask-length, and has a priority of 10.

```
Device(config)# interface gigabitethernet1/0/2
Device(config-if)# ip address 172.21.24.18 255.255.255.0
Device(config-if)# ip pim sparse-mode
Device(config-if)# ip pim bsr-candidate gigabitethernet1/0/2 30 10
```

Example: Configuring Candidate RPs

This example shows how to configure the device to advertise itself as a candidate RP to the BSR in its PIM domain. Standard access list number 4 specifies the group prefix associated with the RP that has the address identified by a port. That RP is responsible for the groups with the prefix 239.

```
Device(config)# ip pim rp-candidate gigabitethernet1/0/2 group-list 4
Device(config)# access-list 4 permit 239.0.0.0 0.255.255.255
```

Additional References

Related Documents

Related Topic	Document Title
For complete syntax and usage information for the commands used in this chapter.	<i>IP Multicast Command Reference, Cisco IOS Release 15.2(2)E (Catalyst 2960-XR Switch)</i>
Cisco IOS IP SLAs commands	Cisco IOS IP Multicast Command Reference

Error Message Decoder

Description	Link
To help you research and resolve system error messages in this release, use the Error Message Decoder tool.	https://www.cisco.com/cgi-bin/Support/Errordecoder/index.cgi

Standards and RFCs

Standard/RFC	Title
PIM is defined in RFC 4601 and in these Internet Engineering Task Force (IETF) Internet drafts.	<ul style="list-style-type: none"> • <i>Protocol Independent Multicast (PIM): Motivation and Architecture</i> • <i>Protocol Independent Multicast (PIM), Dense Mode Protocol Specification</i> • <i>Protocol Independent Multicast (PIM), Sparse Mode Protocol Specification</i> • <i>draft-ietf-idmr-igmp-v2-06.txt, Internet Group Management Protocol, Version 2</i> • <i>draft-ietf-pim-v2-dm-03.txt, PIM Version 2 Dense Mode</i>

MIBs

MIB	MIBs Link
All the supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
<p>The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies.</p> <p>To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds.</p> <p>Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.</p>	http://www.cisco.com/support

Feature History and Information for PIM

Release	Modification
Cisco IOS Release 15.0(2)EX1	This feature was introduced.



CHAPTER 17

Configuring HSRP Aware PIM

- [HSRP Aware PIM, on page 245](#)

HSRP Aware PIM

This module describes how to configure the HSRP Aware PIM feature for enabling multicast traffic to be forwarded through the Hot Standby Router Protocol (HSRP) active router (AR), allowing Protocol Independent Multicast (PIM) to leverage HSRP redundancy, avoid potential duplicate traffic, and enable failover.

Restrictions for HSRP Aware PIM

- HSRP IPv6 is not supported.
- Stateful failover is not supported. During PIM stateless failover, the HSRP group's virtual IP address transfers to the standby router but no mrouting state information is transferred. PIM listens and responds to state change events and creates mroute states upon failover.
- The maximum number of HSRP groups that can be tracked by PIM on each interface is 16.
- The redundancy priority for a PIM DR must be greater than the configured or default value (1) of the PIM DR priority on any device for which the same HSRP group is enabled or the HSRP Active will fail to win the DR election.
- Dense mode is not supported.
- HSRP address as PIM RP is not supported. HSRP aware PIM is for coordinating PIM DR election and HSRP primary election.

Information About HSRP Aware PIM

HSRP

Hot Standby Router Protocol (HSRP) is a Cisco proprietary redundancy protocol for establishing a fault-tolerant default gateway.

The protocol establishes a framework between network devices in order to achieve default gateway failover if the primary gateway becomes inaccessible. By sharing an IP address and a MAC (Layer 2) address, two or more devices can act as a single virtual router. The members of a virtual router group continually exchange

status messages and one device can assume the routing responsibility of another, should it go out of commission for either planned or unplanned reasons. Hosts continue to forward IP packets to a consistent IP and MAC address, and the changeover of devices doing the routing is transparent.

HSRP is useful for hosts that do not support a router discovery protocol and cannot switch to a new device when their selected device reloads or loses power. Because existing TCP sessions can survive the failover, this protocol also provides a more transparent recovery for hosts that dynamically choose a next hop for routing IP traffic.

When HSRP is configured on a network segment, it provides a virtual MAC address and an IP address that is shared among a group of devices running HSRP. The address of this HSRP group is referred to as the virtual IP address. One of these devices is selected by the protocol to be the active router (AR). The AR receives and routes packets destined for the MAC address of the group.

HSRP uses a priority mechanism to determine which HSRP configured device is to be the default AR. To configure a device as the AR, you assign it a priority that is higher than the priority of all the other HSRP-configured devices. The default priority is 100, so if you configure just one device to have a higher priority, that device will be the default AR.

Devices that are running HSRP send and receive multicast User Datagram Protocol (UDP)-based hello messages to detect device failure and to designate active and standby devices. When the AR fails to send a hello message within a configurable period of time, the standby device with the highest priority becomes the AR. The transition of packet forwarding functions between devices is completely transparent to all hosts on the network.

You can configure multiple Hot Standby groups on an interface, thereby making fuller use of redundant devices and load sharing.

HSRP is not a routing protocol as it does not advertise IP routes or affect the routing table in any way.

HSRP has the ability to trigger a failover if one or more interfaces on the device fail. This can be useful for dual branch devices each with a single serial link back to the head end. If the serial link of the primary device goes down, the backup device takes over the primary functionality and thus retains connectivity to the head end.

HSRP Aware PIM

Protocol Independent Multicast (PIM) has no inherent redundancy capabilities and its operation is completely independent of Hot Standby Router Protocol (HSRP) group states. As a result, IP multicast traffic is forwarded not necessarily by the same device as is elected by HSRP. The HSRP Aware PIM feature provides consistent IP multicast forwarding in a redundant network with virtual routing groups enabled.

HSRP Aware PIM enables multicast traffic to be forwarded through the HSRP active router (AR), allowing PIM to leverage HSRP redundancy, avoid potential duplicate traffic, and enable failover, depending on the HSRP states in the device. The PIM designated router (DR) runs on the same gateway as the HSRP AR and maintains mroute states.

In a multiaccess segment (such as LAN), PIM DR election is unaware of the redundancy configuration, and the elected DR and HSRP AR may not be the same router. In order to ensure that the PIM DR is always able to forward PIM Join/Prune message towards RP or FHR, the HSRP AR becomes the PIM DR (if there is only one HSRP group). PIM is responsible for adjusting DR priority based on the group state. When a failover occurs, multicast states are created on the new AR elected by the HSRP group and the AR assumes responsibility for the routing and forwarding of all the traffic addressed to the HSRP virtual IP address.

With HSRP Aware PIM enabled, PIM sends an additional PIM Hello message using the HSRP virtual IP addresses as the source address for each active HSRP group when a device becomes HSRP Active. The PIM

Hello will carry a new GenID in order to trigger other routers to respond to the failover. When a downstream device receives this PIM Hello, it will add the virtual address to its PIM neighbor list. The new GenID carried in the PIM Hello will trigger downstream routers to resend PIM Join messages towards the virtual address. Upstream routers will process PIM Join/Prunes (J/P) based on HSRP group state.

If the J/P destination matches the HSRP group virtual address and if the destination device is in HSRP active state, the new AR processes the PIM Join because it is now the acting PIM DR. This allows all PIM Join/Prunes to reach the HSRP group virtual address and minimizes changes and configurations at the downstream routers side.

The IP routing service utilizes the existing virtual routing protocol to provide basic stateless failover services to client applications, such as PIM. Changes in the local HSRP group state and standby router responsibility are communicated to interested client applications. Client applications may build on top of IRS to provide stateful or stateless failover. PIM, as an HSRP client, listens to the state change notifications from HSRP and automatically adjusts the priority of the PIM DR based on the HSRP state. The PIM client also triggers communication between upstream and downstream devices upon failover in order to create an mroute state on the new AR.

How to Configure HSRP Aware PIM

Configuring an HSRP Group on an Interface

Before you begin

- IP multicast must already be configured on the device.
- PIM must already be configured on the interface.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *type number* [*name-tag*]
4. **ip address** *ip-address mask*
5. **standby** [*group-number*] **ip** [*ip-address* [*secondary*]]
6. **standby** [*group-number*] **timers** [**msec**] *hellotime* [**msec**] *holdtime*
7. **standby** [*group-number*] **priority** *priority*
8. **standby** [*group-number*] **name** *group-name*
9. **end**
10. **show standby** [*type number* [*group*]] [**all** | **brief**]

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.

	Command or Action	Purpose
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	interface <i>type number</i> [<i>name-tag</i>] Example: Device(config)# interface ethernet 0/0	Specifies an interface to be configured and enters interface configuration mode.
Step 4	ip address <i>ip-address mask</i> Example: Device(config-if)# ip address 10.0.0.2 255.255.255.0	Sets a primary or secondary IP address for an interface.
Step 5	standby [<i>group-number</i>] ip [<i>ip-address</i> [<i>secondary</i>]] Example: Device(config-if)# standby 1 ip 192.0.2.99	Activates HSRP and defines an HSRP group.
Step 6	standby [<i>group-number</i>] timers [<i>msec</i>] <i>hellotime</i> [<i>msec</i>] <i>holdtime</i> Example: Device(config-if)# standby 1 timers 5 15	(Optional) Configures the time between hello packets and the time before other devices declare an HSRP active or standby router to be down.
Step 7	standby [<i>group-number</i>] priority <i>priority</i> Example: Device(config-if)# standby 1 priority 120	(Optional) Assigns the HSRP priority to be used to help select the HSRP active and standby routers.
Step 8	standby [<i>group-number</i>] name <i>group-name</i> Example: Device(config-if)# standby 1 name HSRP1	(Optional) Defines a name for the HSRP group. Note We recommend that you always configure the standby ip name command when configuring an HSRP group to be used for HSRP Aware PIM.
Step 9	end Example: Device(config-if)# end	Returns to privileged EXEC mode.
Step 10	show standby [<i>type number</i> [<i>group</i>]] [all brief] Example: Device# show standby	Displays HSRP group information for verifying the configuration.

Configuring PIM Redundancy

Before you begin

The HSRP group must already be configured on the interface. See the “Configuring an HSRP Group on an Interface” section.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *type number* [*name-tag*]
4. **ip address** *ip-address mask*
5. **ip pim redundancy group dr-priority** *priority*
6. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	interface <i>type number</i> [<i>name-tag</i>] Example: Device(config)# interface ethernet 0/0	Specifies an interface to be configured and enters interface configuration mode.
Step 4	ip address <i>ip-address mask</i> Example: Device(config-if)# ip address 10.0.0.2 255.255.255.0	Sets a primary or secondary IP address for an interface.
Step 5	ip pim redundancy group dr-priority <i>priority</i> Example: Device(config-if)# ip pim redundancy HSRP1 dr-priority 60	Enables PIM redundancy and assigns a redundancy priority value to the active PIM designated router (DR). <ul style="list-style-type: none"> • Because HSRP group names are case sensitive, the value of the <i>group</i> argument must match the group name configured by using the standby ip name command. • The redundancy priority for a PIM DR must be greater than the configured or default value (1) of the PIM DR priority on any device for which the same HSRP group is enabled.

	Command or Action	Purpose
Step 6	end Example: Device(config-if)# end	Returns to privileged EXEC mode.

Configuration Examples for HSRP Aware PIM

Example: Configuring an HSRP Group on an Interface

```
interface ethernet 0/0
 ip address 10.0.0.2 255.255.255.0
 standby 1 ip 192.0.2.99
 standby 1 timers 5 15
 standby 1 priority 120
 standby 1 name HSRP1
!
```

Example: Configuring PIM Redundancy

```
interface ethernet 0/0
 ip address 10.0.0.2 255.255.255.0
 ip pim redundancy HSRP1 dr-priority 60
!
```

Additional References for HSRP Aware PIM

Related Documents

Related Topic	Document Title
IP multicast commands	Cisco IOS IP Multicast Command Reference
HSRP commands	First Hop Redundancy Protocol Command Reference

Standards and RFCs

Standard/RFC	Title
RFC 2281	<i>Cisco Hot Standby Router Protocol (HSRP)</i>

MIBs

MIB	MIBs Link
No new or modified MIBs are supported by this feature, and support for existing MIBs has not been modified by this feature.	To locate and download MIBs for selected platforms, Cisco software releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

Feature Information for HSRP Aware PIM

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.



CHAPTER 18

Configuring VRRP Aware PIM

- [VRRP Aware PIM, on page 253](#)

VRRP Aware PIM

The Virtual Router Redundancy Protocol (VRRP) eliminates the single point of failure inherent in the static default routed environment. VRRP is an election protocol that dynamically assigns responsibility for one or more virtual routers to the VRRP routers on a LAN, allowing several routers on a multi access link to utilize the same virtual IP address.

VRRP Aware PIM is a redundancy mechanism for the Protocol Independent Multicast (PIM) to interoperate with VRRP. It allows PIM to track VRRP state and to preserve multicast traffic upon fail over in a redundant network with virtual routing groups enabled.

This module explains how to configure VRRP Aware PIM in a network.

Restrictions for VRRP Aware PIM

- Only PIM sparse mode (SM) and source specific multicast (SSM) modes are supported. Bidirectional (BiDir) PIM is not supported.
- PIM interoperability with Hot Standby Router Protocol (HSRP) IPv6 is not supported.
- PIM tracks only one virtual group, either Virtual Router Redundancy Protocol (VRRP) or HSRP, per interface.
- VRRP Aware PIM is not supported on a Transit network. PIM redundancy enabled interface does not support the PIM joining the network from down stream.

Information About VRRP Aware PIM

Overview of VRRP Aware PIM

Virtual Router Redundancy Protocol (VRRP) is a redundancy protocol for establishing a fault-tolerant default gateway. The protocol establishes a framework between network devices in order to achieve default gateway failover if the primary gateway becomes inaccessible.

Protocol Independent Multicast (PIM) has no inherent redundancy capabilities and its operation is completely independent of VRRP group states. As a result, IP multicast traffic is forwarded not necessarily by the same

device as is elected by VRRP. The VRRP Aware PIM feature provides consistent IP multicast forwarding in a redundant network with virtual routing groups enabled.

In a multi-access segment (such as LAN), PIM designated router (DR) election is unaware of the redundancy configuration, and the elected DR and VRRP primary router (MR) may not be the same router. In order to ensure that the PIM DR is always able to forward PIM Join/Prune message towards RP or FHR, the VRRP MR becomes the PIM DR (if there is only one VRRP group). PIM is responsible for adjusting DR priority based on the group state. When a fail over occurs, multicast states are created on the new MR elected by the VRRP group and the MR assumes responsibility for the routing and forwarding of all the traffic addressed to the VRRP virtual IP address. This ensures the PIM DR runs on the same gateway as the VRRP MR and maintains mroute states. It enables multicast traffic to be forwarded through the VRRP MR, allowing PIM to leverage VRRP redundancy, avoid potential duplicate traffic, and enable fail over, depending on the VRRP states in the device.

Virtual Router Redundancy Service (VRRS) provides public APIs for a client to communicate with VRRP. VRRP Aware PIM is a feature of VRRS that supports VRRPv3 (unified VRRP) in both IPv4 and IPv6.

PIM, as a VRRS client, uses the VRRS client API to obtain generic First Hop Redundancy Protocol (FHRP) state and configuration information in order to provide multicast redundancy functionalities.

PIM performs the following as a VRRS client:

- Listens to state change and update notification from VRRS server (i.e., VRRP).
- Automatically adjust PIM DR priority based on VRRP state.
- Upon VRRP fail over, PIM receives state change notification from VRRS for the tracked VRRP group and ensures traffic is forwarded through VRRP MR.

How to Configure VRRP Aware PIM

Configuring VRRP Aware PIM

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **fhrp version vrrp *version***
4. **interface *type number***
5. **ip address *address* {*primary* |*secondary*}**
6. **vrrp *group id* address-family ipv4**
7. **vrrs leader *group name***
8. **vrrp *group id* ip *ip address* {*primary* |*secondary*}**
9. **exit**
10. **interface *type number***
11. **ip pim redundancy *group name* vrrp dr-priority *priority-value***
12. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Router# configure terminal	Enters global configuration mode.
Step 3	fhrp version vrrp version Example: Device(config)# fhrp version vrrp v3	Enables the ability to configure VRRPv3 and VRRS.
Step 4	interface type number Example: Device(config)# interface Ethernet0/0	Specifies an interface to be configured and enters interface configuration mode.
Step 5	ip address address {primary secondary} Example: Device(config-if)# ip address 192.0.2.2	Specifies a primary or secondary address for the VRRP group.
Step 6	vrrp group id address-family ipv4 Example: Device(config-if)# vrrp 1 address-family ipv4	Creates a VRRP group and enters VRRP configuration mode.
Step 7	vrrs leader group name Example: Device(config-if-vrrp)# vrrs leader VRRP1	Enables community and (or) extended community exchange with the specified neighbor.
Step 8	vrrp group id ip ip address {primary secondary} Example: Device(config-if-vrrp)# vrrp 1 ip 10.1.6.1	Exits address family configuration mode and returns to router configuration mode.
Step 9	exit Example: Device(config-if-vrrp)# exit	Exits VRRP configuration mode and returns to global configuration mode.

	Command or Action	Purpose
Step 10	interface <i>type number</i> Example: Device(config)# interface Ethernet0/0	Specifies an interface to be configured and enters interface configuration mode.
Step 11	ip pim redundancy <i>group name</i> vrrp dr-priority <i>priority-value</i> Example: Device(config-if)# ip pim redundancy VRRP1 vrrp dr-priority 90	sets the priority for which a router is elected as the designated router (DR). <ul style="list-style-type: none"> The redundancy dr-priority value should be same on all routers that are enabled with VRRP Aware PIM feature.
Step 12	end Example: Device(config-if)# end	Exits interface configuration mode and returns to privileged EXEC mode.

Configuration Examples for VRRP Aware PIM

Example: VRRP Aware PIM

```

conf terminal
 fhrp version vrrp v3
 interface Ethernet0/0
 ip address 192.0.2.2
 vrrp 1 address-family ipv4

 vrrp 1 ip 10.1.6.1

 vrrs leader VRRP1
 interface Ethernet0/0
 ip pim redundancy VRRP1 vrrp dr-priority 90
 !

```

Additional References for VRRP Aware PIM

Related Documents

Related Topic	Document Title
IP multicast commands	Cisco IOS IP Multicast Command Reference
Configuring VRRP	First Hop Redundancy Protocols Configuration Guide

Related Topic	Document Title
IP multicast PIM	IP Multicast: PIM Configuration Guide

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html



CHAPTER 19

Configuring SSM

- [Prerequisites for Configuring SSM, on page 259](#)
- [Restrictions for Configuring SSM, on page 259](#)
- [Information About SSM and SSM Mapping, on page 261](#)
- [How to Configure SSM and SSM Mapping, on page 266](#)
- [Monitoring SSM and SSM Mapping, on page 274](#)
- [Configuration Examples for SSM and SSM Mapping, on page 275](#)
- [Additional References, on page 279](#)
- [Feature History and Information for SSM, on page 280](#)

Prerequisites for Configuring SSM

The following are the prerequisites for configuring source-specific multicast (SSM) and SSM mapping:

- Before you configure SSM mapping, you must perform the following tasks:
 - Enable IP multicast routing.
 - Enable PIM sparse mode.
 - Configure SSM.
- Before you configure static SSM mapping, you must configure access control lists (ACLs) that define the group ranges to be mapped to source addresses.
- Before you can configure and use SSM mapping with DNS lookups, you need to add records to a running DNS server. If you do not already have a DNS server running, you need to install one.



Note You can use a product such as *Cisco Network Registrar* to add records to a running DNS server.

Restrictions for Configuring SSM

The following are the restrictions for configuring SSM:

- To run SSM with IGMPv3, SSM must be supported in the Cisco IOS router, the host where the application is running, and the application itself.
- Existing applications in a network predating SSM will not work within the SSM range unless they are modified to support (S, G) channel subscriptions. Therefore, enabling SSM in a network may cause problems for existing applications if they use addresses within the designated SSM range.
- IGMP Snooping—IGMPv3 uses new membership report messages that might not be correctly recognized by older IGMP snooping devices.
- Address management is still necessary to some degree when SSM is used with Layer 2 switching mechanisms. Cisco Group Management Protocol (CGMP), IGMP snooping, or Router-Port Group Management Protocol (RGMP) support only group-specific filtering, not (S, G) channel-specific filtering. If different receivers in a switched network request different (S, G) channels sharing the same group, they do not benefit from these existing mechanisms. Instead, both receivers receive all (S, G) channel traffic and filter out the unwanted traffic on input. Because SSM can re-use the group addresses in the SSM range for many independent applications, this situation can lead to decreased traffic filtering in a switched network. For this reason, it is important to use random IP addresses from the SSM range for an application to minimize the chance for re-use of a single address within the SSM range between different applications. For example, an application service providing a set of television channels should, even with SSM, use a different group for each television (S, G) channel. This setup guarantees that multiple receivers to different channels within the same application service never experience traffic aliasing in networks that include Layer 2 devices.
- In PIM-SSM, the last hop router will continue to periodically send (S, G) join messages if appropriate (S, G) subscriptions are on the interfaces. Therefore, as long as receivers send (S, G) subscriptions, the shortest path tree (SPT) state from the receivers to the source will be maintained, even if the source is not sending traffic for longer periods of time (or even never).

The opposite situation occurs with PIM-SM, where (S, G) state is maintained only if the source is sending traffic and receivers are joining the group. If a source stops sending traffic for more than 3 minutes in PIM-SM, the (S, G) state is deleted and only reestablished after packets from the source arrive again through the RPT (rendezvous point tree). Because no mechanism in PIM-SSM notifies a receiver that a source is active, the network must maintain the (S, G) state in PIM-SSM as long as receivers are requesting receipt of that channel.

The following are the restrictions for configuring SSM mapping:

- The SSM Mapping feature does not share the benefit of full SSM. SSM mapping takes a group G join from a host and identifies this group with an application associated with one or more sources, therefore, it can only support one such application per group G. Nevertheless, full SSM applications may still share the same group also used in SSM mapping.
- Enable IGMPv3 with care on the last hop router when you rely solely on SSM mapping as a transition solution for full SSM. When you enable both SSM mapping and IGMPv3 and the hosts already support IGMPv3 (but not SSM), the hosts send IGMPv3 group reports. SSM mapping does not support these IGMPv3 group reports, and the router does not correctly associate sources with these reports.

Information About SSM and SSM Mapping

SSM Components Overview

SSM is a datagram delivery model that best supports one-to-many applications, also known as broadcast applications. It is an extension of IP multicast in which datagram traffic is forwarded to receivers from only those multicast sources that the receivers have explicitly joined. For multicast groups configured for SSM, only SSM distribution trees (no shared trees) are created.

SSM is a core networking technology for Cisco's implementation of IP multicast solutions targeted for audio and video broadcast application environments and is described in RFC 3569. The following components together support the implementation of SSM:

- Protocol Independent Multicast source-specific mode (PIM-SSM)
- Internet Group Management Protocol Version 3 (IGMPv3)

Protocol Independent Multicast (PIM) SSM, or PIM-SSM, is the routing protocol that supports the implementation of SSM and is derived from PIM sparse mode (PIM-SM). IGMP is the Internet Engineering Task Force (IETF) standards track protocol used for hosts to signal multicast group membership to routers. IGMP Version 3 supports source filtering, which is required for SSM. For SSM to run with IGMPv3, SSM must be supported in the router, the host where the application is running, and the application itself.

How SSM Differs from Internet Standard Multicast

The standard IP multicast infrastructure in the Internet and many enterprise intranets is based on the PIM-SM protocol and Multicast Source Discovery Protocol (MSDP). These protocols have proved to be reliable, extensive, and efficient. However, they are bound to the complexity and functionality limitations of the Internet Standard Multicast (ISM) service model. For example, with ISM, the network must maintain knowledge about which hosts in the network are actively sending multicast traffic. With SSM, this information is provided by receivers through the source addresses relayed to the last-hop devices by IGMPv3. SSM is an incremental response to the issues associated with ISM and is intended to coexist in the network with the protocols developed for ISM. In general, SSM provides IP multicast service for applications that utilize SSM.

ISM service is described in RFC 1112. This service consists of the delivery of IP datagrams from any source to a group of receivers called the multicast host group. The datagram traffic for the multicast host group consists of datagrams with an arbitrary IP unicast source address *S* and the multicast group address *G* as the IP destination address. Systems will receive this traffic by becoming members of the host group. Membership in a host group simply requires signaling the host group through IGMP Version 1, 2, or 3.

In SSM, delivery of datagrams is based on (*S, G*) channels. Traffic for one (*S, G*) channel consists of datagrams with an IP unicast source address *S* and the multicast group address *G* as the IP destination address. Systems will receive this traffic by becoming members of the (*S, G*) channel. In both SSM and ISM, no signaling is required to become a source. However, in SSM, receivers must subscribe or unsubscribe to (*S, G*) channels to receive or not receive traffic from specific sources. In other words, receivers can receive traffic only from (*S, G*) channels to which they are subscribed, whereas in ISM, receivers need not know the IP addresses of sources from which they receive their traffic. The proposed standard approach for channel subscription signaling utilizes IGMP INCLUDE mode membership reports, which are supported only in IGMP Version 3.

SSM can coexist with the ISM service by applying the SSM delivery model to a configured subset of the IP multicast group address range. The Internet Assigned Numbers Authority (IANA) has reserved the address range from 232.0.0.0 through 232.255.255.255 for SSM applications and protocols. The software allows SSM configuration for an arbitrary subset of the IP multicast address range from 224.0.0.0 through 239.255.255.255. When an SSM range is defined, an existing IP multicast receiver application will not receive any traffic when it tries to use addresses in the SSM range unless the application is modified to use explicit (S, G) channel subscription or is SSM-enabled through a URL Rendezvous Directory (URD).

SSM Operations

An established network in which IP multicast service is based on PIM-SM can support SSM services. SSM can also be deployed alone in a network without the full range of protocols that are required for interdomain PIM-SM. That is, SSM does not require an RP, so there is no need for an RP mechanism such as Auto-RP, MSDP, or bootstrap router (BSR).

If SSM is deployed in a network that is already configured for PIM-SM, then only the last-hop routers must be upgraded to a software image that supports SSM. Routers that are not directly connected to receivers do not have to upgrade to a software image that supports SSM. In general, these non-last-hop routers must only run PIM-SM in the SSM range. They may need additional access control configuration to suppress MSDP signaling, registering, or PIM-SM shared-tree operations from occurring within the SSM range.

The SSM mode of operation is enabled by configuring the SSM range using the `ip pim ssm` global configuration command. This configuration has the following effects:

- For groups within the SSM range, (S, G) channel subscriptions are accepted through IGMPv3 INCLUDE mode membership reports.
- PIM operations within the SSM range of addresses change to PIM-SSM, a mode derived from PIM-SM. In this mode, only PIM (S, G) Join and Prune messages are generated by the router. Incoming messages related to rendezvous point tree (RPT) operations are ignored or rejected, and incoming PIM register messages are immediately answered with Register-Stop messages. PIM-SSM is backward-compatible with PIM-SM unless a router is a last-hop router. Therefore, routers that are not last-hop routers can run PIM-SM for SSM groups (for example, if they do not yet support SSM).
- For groups within the SSM range, no MSDP Source-Active (SA) messages within the SSM range will be accepted, generated, or forwarded.

IGMPv3 Host Signaling

IGMPv3 is the third version of the IETF standards track protocol in which hosts signal membership to last-hop routers of multicast groups. IGMPv3 introduces the ability for hosts to signal group membership that allows filtering capabilities with respect to sources. A host can signal either that it wants to receive traffic from all sources sending to a group except for some specific sources (a mode called EXCLUDE) or that it wants to receive traffic only from some specific sources sending to the group (a mode called INCLUDE).

IGMPv3 can operate with both ISM and SSM. In ISM, both EXCLUDE and INCLUDE mode reports are accepted by the last-hop router. In SSM, only INCLUDE mode reports are accepted by the last-hop router.

Benefits of Source Specific Multicast

IP Multicast Address Management Not Required

In the ISM service, applications must acquire a unique IP multicast group address because traffic distribution is based only on the IP multicast group address used. If two applications with different sources and receivers use the same IP multicast group address, then receivers of both applications will receive traffic from the senders of both applications. Even though the receivers, if programmed appropriately, can filter out the unwanted traffic, this situation would cause generally unacceptable levels of unwanted traffic.

Allocating a unique IP multicast group address for an application is still a problem. Most short-lived applications use mechanisms like Session Description Protocol (SDP) and Session Announcement Protocol (SAP) to get a random address, a solution that does not work well with a rising number of applications in the Internet. The best current solution for long-lived applications is described in RFC 2770, but this solution suffers from the restriction that each autonomous system is limited to only 255 usable IP multicast addresses.

In SSM, traffic from each source is forwarded between routers in the network independent of traffic from other sources. Thus different sources can reuse multicast group addresses in the SSM range.

Denial of Service Attacks from Unwanted Sources Inhibited

In SSM, multicast traffic from each individual source will be transported across the network only if it was requested (through IGMPv3, IGMP v3lite, or URD memberships) from a receiver. In contrast, ISM forwards traffic from any active source sending to a multicast group to all receivers requesting that multicast group. In Internet broadcast applications, this ISM behavior is highly undesirable because it allows unwanted sources to easily disturb the actual Internet broadcast source by simply sending traffic to the same multicast group. This situation depletes bandwidth at the receiver side with unwanted traffic and thus disrupts the undisturbed reception of the Internet broadcast. In SSM, this type of denial of service (DoS) attack cannot be made by simply sending traffic to a multicast group.

Easy to Install and Manage

SSM is easy to install and provision in a network because it does not require the network to maintain which active sources are sending to multicast groups. This requirement exists in ISM (with IGMPv1, IGMPv2, or IGMPv3).

The current standard solutions for ISM service are PIM-SM and MSDP. Rendezvous point (RP) management in PIM-SM (including the necessity for Auto-RP or BSR) and MSDP is required only for the network to learn about active sources. This management is not necessary in SSM, which makes SSM easier than ISM to install and manage, and therefore easier than ISM to operationally scale in deployment. Another factor that contributes to the ease of installation of SSM is the fact that it can leverage preexisting PIM-SM networks and requires only the upgrade of last hop routers to support IGMPv3, IGMP v3lite, or URD.

Ideal for Internet Broadcast Applications

The three benefits previously described make SSM ideal for Internet broadcast-style applications for the following reasons:

- The ability to provide Internet broadcast services through SSM without the need for unique IP multicast addresses allows content providers to easily offer their service (IP multicast address allocation has been a serious problem for content providers in the past).
- The prevention against DoS attacks is an important factor for Internet broadcast services because, with their exposure to a large number of receivers, they are the most common targets for such attacks.

- The ease of installation and operation of SSM makes it ideal for network operators, especially in those cases where content needs to be forwarded between multiple independent PIM domains (because there is no need to manage MSDP for SSM between PIM domains).

SSM Mapping Overview

SSM mapping supports SSM transition when supporting SSM on the end system is impossible or unwanted due to administrative or technical reasons. Using SSM to deliver live streaming video to legacy STBs that do not support IGMPv3 or for applications that do not use the IGMPv3 host stack is a typical application of SSM mapping.

In a typical STB deployment, each TV channel uses one separate IP multicast group and has one active server host sending the TV channel. A single server may of course send multiple TV channels, but each to a different group. In this network environment, if a router receives an IGMPv1 or IGMPv2 membership report for a particular group G, the report implicitly addresses the well-known TV server for the TV channel associated with the multicast group.

SSM mapping introduces a means for the last hop router to discover sources sending to groups. When SSM mapping is configured, if a router receives an IGMPv1 or IGMPv2 membership report for a particular group G, the router translates this report into one or more (S, G) channel memberships for the well-known sources associated with this group.

When the router receives an IGMPv1 or IGMPv2 membership report for group G, the router uses SSM mapping to determine one or more source IP addresses for group G. SSM mapping then translates the membership report as an IGMPv3 report INCLUDE (G, [S1, G], [S2, G]...[Sn, G]) and continues as if it had received an IGMPv3 report. The router then sends out PIM joins toward (S1, G) to (Sn, G) and continues to be joined to these groups as long as it continues to receive the IGMPv1 or IGMPv2 membership reports and as long as it continues to receive the IGMPv1 or IGMPv2 membership reports, and the SSM mapping for the group remains the same. SSM mapping, thus, enables you to leverage SSM for video delivery to legacy STBs that do not support IGMPv3 or for applications that do not take advantage of the IGMPv3 host stack.

SSM mapping enables the last hop router to determine the source addresses either by a statically configured table on the router or by consulting a DNS server. When the statically configured table is changed, or when the DNS mapping changes, the router will leave the current sources associated with the joined groups.

Static SSM Mapping

SSM static mapping enables you to configure the last hop router to use a static map to determine the sources sending to groups. Static SSM mapping requires that you configure access lists (ACLs) to define group ranges. The groups permitted by those ACLs then can be mapped to sources using the **ip igmp static ssm-map** global configuration command.

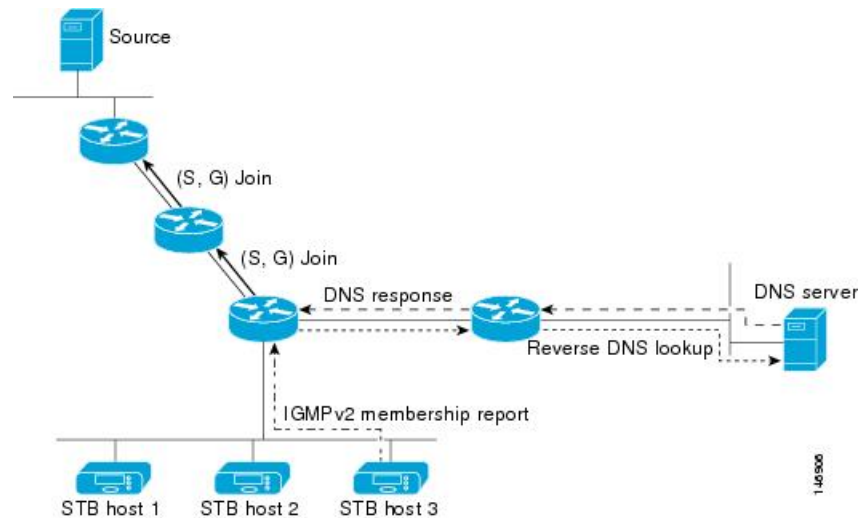
You can configure static SSM mapping in smaller networks when a DNS is not needed or to locally override DNS mappings that may be temporarily incorrect. When configured, static SSM mappings take precedence over DNS mappings.

DNS-Based SSM Mapping

DNS-based SSM mapping enables you to configure the last hop router to perform a reverse DNS lookup to determine sources sending to groups (see the figure below). When DNS-based SSM mapping is configured, the router constructs a domain name that includes the group address G and performs a reverse lookup into the DNS. The router looks up IP address resource records (IP A RRs) to be returned for this constructed domain

name and uses the returned IP addresses as the source addresses associated with this group. SSM mapping supports up to 20 sources for each group. The router joins all sources configured for a group.

Figure 15: DNS-Based SSM-Mapping



The SSM mapping mechanism that enables the last hop router to join multiple sources for a group can be used to provide source redundancy for a TV broadcast. In this context, the redundancy is provided by the last hop router using SSM mapping to join two video sources simultaneously for the same TV channel. However, to prevent the last hop router from duplicating the video traffic, it is necessary that the video sources utilize a server-side switchover mechanism where one video source is active while the other backup video source is passive. The passive source waits until an active source failure is detected before sending the video traffic for the TV channel. The server-side switchover mechanism, thus, ensures that only one of the servers is actively sending the video traffic for the TV channel.

To look up one or more source addresses for a group G that includes G1, G2, G3, and G4, the following DNS resource records (RRs) must be configured on the DNS server:

G4.G3.G2.G1 [<i>multicast-domain</i>] [<i>timeout</i>]	IN A <i>source-address-1</i>
	IN A <i>source-address-2</i>
	IN A <i>source-address-n</i>

The *multicast-domain* argument is a configurable DNS prefix. The default DNS prefix is `in-addr.arpa`. You should only use the default prefix when your installation is either separate from the internet or if the group names that you map are global scope group addresses (RFC 2770 type addresses that you configure for SSM) that you own.

The *timeout* argument configures the length of time for which the router performing SSM mapping will cache the DNS lookup. This argument is optional and defaults to the timeout of the zone in which this entry is configured. The timeout indicates how long the router will keep the current mapping before querying the DNS server for this group. The timeout is derived from the cache time of the DNS RR entry and can be configured for each group/source entry on the DNS server. You can configure this time for larger values if you want to minimize the number of DNS queries generated by the router. Configure this time for a low value if you want to be able to quickly update all routers with new source addresses.



Note Refer to your DNS server documentation for more information about configuring DNS RRs.

To configure DNS-based SSM mapping in the software, you must configure a few global commands but no per-channel specific configuration is needed. There is no change to the configuration for SSM mapping if additional channels are added. When DNS-based SSM mapping is configured, the mappings are handled entirely by one or more DNS servers. All DNS techniques for configuration and redundancy management can be applied to the entries needed for DNS-based SSM mapping.

SSM Mapping Benefits

- The SSM Mapping feature provides almost the same ease of network installation and management as a pure SSM solution based on IGMPv3. Some additional configuration is necessary to enable SSM mapping.
- The SSM benefit of inhibition of DoS attacks applies when SSM mapping is configured. When SSM mapping is configured the only segment of the network that may still be vulnerable to DoS attacks are receivers on the LAN connected to the last hop router. Since those receivers may still be using IGMPv1 and IGMPv2, they are vulnerable to attacks from unwanted sources on the same LAN. SSM mapping, however, does protect those receivers (and the network path leading towards them) from multicast traffic from unwanted sources anywhere else in the network.
- Address assignment within a network using SSM mapping needs to be coordinated, but it does not need assignment from outside authorities, even if the content from the network is to be transited into other networks.

How to Configure SSM and SSM Mapping

Configuring SSM

Follow these steps to configure SSM:

This procedure is optional.

Before you begin

If you want to use an access list to define the Source Specific Multicast (SSM) range, configure the access list before you reference the access list in the **ip pim ssm** command.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip pim ssm [default | range *access-list*]**
4. **interface *type number***
5. **ip pim {sparse-mode | sparse-dense-mode}**
6. **ip igmp version 3**
7. **end**
8. **show running-config**

9. copy running-config startup-config

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	ip pim ssm [default range access-list] Example: Device(config)# ip pim ssm range 20	Defines the SSM range of IP multicast addresses.
Step 4	interface type number Example: Device(config)# interface gigabitethernet 1/0/1	Selects an interface that is connected to hosts on which IGMPv3 can be enabled, and enters the interface configuration mode.
Step 5	ip pim {sparse-mode sparse-dense-mode} Example: Device(config-if)# ip pim sparse-mode	Enables PIM on an interface.
Step 6	ip igmp version 3 Example: Device(config-if)# ip igmp version 3	Enables IGMPv3 on this interface. The default version of IGMP is set to Version 2.
Step 7	end Example: Device(config)# end	Returns to privileged EXEC mode.
Step 8	show running-config Example:	Verifies your entries.

	Command or Action	Purpose
	Device# <code>show running-config</code>	
Step 9	copy running-config startup-config Example: Device# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Configuring SSM Mapping

Configuring Static SSM Mapping

Follow these steps to configure static SSM Mapping:

SUMMARY STEPS

1. `enable`
2. `configure terminal`
3. `ip igmp ssm-map enable`
4. `no ip igmp ssm-map query dns`
5. `ip igmp ssm-map static access-list source-address`
6. `end`
7. `show running-config`
8. `copy running-config startup-config`

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> <code>enable</code>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode.
Step 3	ip igmp ssm-map enable Example: Device(config)# <code>ip igmp ssm-map enable</code>	Enables SSM mapping for groups in the configured SSM range. <p>Note By default, this command enables DNS-based SSM mapping.</p>

	Command or Action	Purpose
Step 4	<p>no ip igmp ssm-map query dns</p> <p>Example:</p> <pre>Device(config)# no ip igmp ssm-map query dns</pre>	<p>(Optional) Disables DNS-based SSM mapping.</p> <p>Note Disable DNS-based SSM mapping if you only want to rely on static SSM mapping. By default, the ip igmp ssm-map command enables DNS-based SSM mapping.</p>
Step 5	<p>ip igmp ssm-map static <i>access-list source-address</i></p> <p>Example:</p> <pre>Device(config)# ip igmp ssm-map static 11 172.16.8.11</pre>	<p>Configures static SSM mapping.</p> <ul style="list-style-type: none"> The ACL supplied for the <i>access-list</i> argument defines the groups to be mapped to the source IP address entered for the <i>source-address</i> argument. <p>Note You can configure additional static SSM mappings. If additional SSM mappings are configured and the router receives an IGMPv1 or IGMPv2 membership report for a group in the SSM range, the device determines the source addresses associated with the group by walking each configured ip igmp ssm-map static command. The device associates up to 20 sources per group.</p> <p>Repeat Step to configure additional static SSM mappings, if required.</p>
Step 6	<p>end</p> <p>Example:</p> <pre>Device(config)# end</pre>	<p>Returns to privileged EXEC mode.</p>
Step 7	<p>show running-config</p> <p>Example:</p> <pre>Device# show running-config</pre>	<p>Verifies your entries.</p>
Step 8	<p>copy running-config startup-config</p> <p>Example:</p> <pre>Device# copy running-config startup-config</pre>	<p>(Optional) Saves your entries in the configuration file.</p>

Configuring DNS-Based SSM Mapping (CLI)

Perform this task to configure the last hop router to perform DNS lookups to learn the IP addresses of sources sending to a group.

Before you begin

- Enable IP multicast routing, enable PIM sparse mode, and configure SSM before performing this task. For more information, see the "Configuring Basic Multicast" module.
- Before you can configure and use SSM mapping with DNS lookups, you need to be able to add records to a running DNS server. If you do not already have a DNS server running, you need to install one.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip igmp ssm-map enable**
4. **ip igmp ssm-map query dns**
5. **ip domain multicast** *domain-prefix*
6. **ipname-server** *server-address1* [*server-address2server-address6*]
7. Repeat Step 6 to configure additional DNS servers for redundancy, if required.
8. **end**
9. **show running-config**
10. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device# enable	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	ip igmp ssm-map enable Example: Device(config)# ip igmp ssm-map enable	Enables SSM mapping for groups in a configured SSM range.
Step 4	ip igmp ssm-map query dns Example: Device(config)# ip igmp ssm-map query dns	(Optional) Enables DNS-based SSM mapping. <ul style="list-style-type: none"> • By default, the ip igmp ssm-map command enables DNS-based SSM mapping. Only the noform of this command is saved to the running configuration. Note Use this command to reenable DNS-based SSM mapping if DNS-based SSM mapping is disabled.
Step 5	ip domain multicast <i>domain-prefix</i> Example:	(Optional) Changes the domain prefix used by the Cisco IOS XE software for DNS-based SSM mapping.

	Command or Action	Purpose
	Device(config)# ip domain multicast ssm-map.cisco.com	<ul style="list-style-type: none"> By default, the software uses the ip-addr.arpa domain prefix.
Step 6	ipname-server <i>server-address1</i> [<i>server-address2server-address6</i>] Example: Device(config)# ip name-server 10.48.81.21	Specifies the address of one or more name servers to use for name and address resolution.
Step 7	Repeat Step 6 to configure additional DNS servers for redundancy, if required.	--
Step 8	end Example: Device(config-if)# end	Returns to privileged EXEC mode.
Step 9	show running-config Example: Device# show running-config	Verifies your entries.
Step 10	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Configuring Static Traffic Forwarding with SSM Mapping

Follow these steps to configure static traffic forwarding with SSM mapping on the last hop router:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **ip igmp static-group** *group-address* **source ssm-map**
5. **end**
6. **show running-config**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example:	Enables privileged EXEC mode. <ul style="list-style-type: none"> Enter your password if prompted.

	Command or Action	Purpose
	Device> enable	
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	interface <i>interface-id</i> Example: Device(config)# interface gigabitethernet 1/0/1	Selects an interface on which to statically forward traffic for a multicast group using SSM mapping, and enters interface configuration mode. Note Static forwarding of traffic with SSM mapping works with either DNS-based SSM mapping or statically configured SSM mapping.
Step 4	ip igmp static-group <i>group-address</i> source ssm-map Example: Device(config-if)# ip igmp static-group 239.1.2.1 source ssm-map	Configures SSM mapping to statically forward a (S, G) channel from the interface. Use this command if you want to statically forward SSM traffic for certain groups. Use DNS-based SSM mapping to determine the source addresses of the channels.
Step 5	end Example: Device(config)# end	Returns to privileged EXEC mode.
Step 6	show running-config Example: Device# show running-config	Verifies your entries.
Step 7	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Verifying SSM Mapping Configuration and Operation

Follow these steps to verify SSM mapping configuration and operation:

SUMMARY STEPS

1. **enable**
2. **show ip igmp ssm-mapping**
3. **show ip igmp ssm-mapping group-address**
4. **show ip igmp groups** [*group-name* | *group-address* | *interface-type interface-number*] [**detail**]
5. **show host**
6. **debug ip igmp group-address**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Device> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	show ip igmp ssm-mapping Example: <pre>Device# show ip igmp ssm-mapping SSM Mapping : Enabled DNS Lookup : Enabled Mcast domain : ssm-map.cisco.com Name servers : 10.0.0.3 10.0.0.4</pre>	(Optional) Displays information about SSM mapping configuration.
Step 3	show ip igmp ssm-mapping group-address Example: <pre>Device# show ip igmp ssm-mapping 232.1.1.4 Group address: 232.1.1.4 Database : DNS DNS name : 4.1.1.232.ssm-map.cisco.com Expire time : 860000 Source list : 172.16.8.5 : 172.16.8.6</pre>	(Optional) Displays the sources that SSM mapping uses for a particular group. The example here shows information about the configured DNS-based SSM mapping. Here the router has used DNS-based mapping to map group 232.1.1.4 to sources 172.16.8.5 and 172.16.8.6. The timeout for this entry is 860000 milliseconds (860 seconds).
Step 4	show ip igmp groups [<i>group-name</i> <i>group-address</i> <i>interface-type interface-number</i>] [detail] Example: <pre>Device# show ip igmp group 232.1.1.4 detail Interface: GigabitEthernet2/0/0 Group: 232.1.1.4 SSM Uptime: 00:03:20 Group mode: INCLUDE Last reporter: 0.0.0.0 CSR Grp Exp: 00:02:59 Group source list: (C - Cisco Src Report, U - URD, R - Remote, S - Static, M - SSM Mapping)</pre>	(Optional) Displays the multicast groups with receivers that are directly connected to the router and that were learned through IGMP. In the example the “M” flag indicates that SSM mapping is configured.

	Command or Action	Purpose
	<pre> Source Address Uptime v3 Exp CSR Exp Fwd Flags 172.16.8.3 00:03:20 stopped 00:02:59 Yes CM 172.16.8.4 00:03:20 stopped 00:02:59 Yes CM 172.16.8.5 00:03:20 stopped 00:02:59 Yes CM 172.16.8.6 00:03:20 stopped 00:02:59 Yes CM </pre>	
Step 5	<p>show host</p> <p>Example:</p> <pre> Device# show host Default domain is cisco.com Name/address lookup uses domain service Name servers are 10.48.81.21 Codes: UN - unknown, EX - expired, OK - OK, ?? - revalidate temp - temporary, perm - permanent NA - Not Applicable None - Not defined Host Port Flags Age Type Address(es) 10.0.0.0.ssm-map.cisco.c None (temp, OK) 0 IP 172.16.8.5 172.16.8.6 172.16.8.3 </pre>	(Optional) Displays the default domain name, the style of name lookup service, a list of name server hosts, and the cached list of hostnames and addresses.
Step 6	<p>debug ip igmp group-address</p> <p>Example:</p> <pre> Device# debug ip igmp IGMP(0): Convert IGMPv2 report (*,232.1.2.3) to IGMPv3 with 2 source(s) using STATIC. Device# debug ip igmp IGMP(0): Convert IGMPv2 report (*,232.1.2.3) to IGMPv3 with 2 source(s) using DNS. Device# debug ip igmp IGMP(0): DNS source lookup failed for (*, 232.1.2.3), IGMPv2 report failed </pre>	<p>(Optional) Displays the IGMP packets received and sent and IGMP host-related events.</p> <p>In the first example, the output indicates that the router is converting an IGMPv2 join for group G into an IGMPv3 join.</p> <p>In the second example, the output indicates that a DNS lookup has succeeded.</p> <p>In the third example, the output indicates that DNS-based SSM mapping is enabled and a DNS lookup has failed:</p>

Monitoring SSM and SSM Mapping

Monitoring SSM

To monitor SSM, use the following commands in privileged EXEC mode, as needed:

Command	Purpose
Device# show ip igmp groups detail	Displays the (S, G) channel subscription through IGMPv3.
Device# show ip mroute	Displays whether a multicast group supports SSM service or whether a source-specific host report was received.

Monitoring SSM Mapping

Use the privileged EXEC commands in the following table to monitor SSM mapping.

Table 33: SSM Mapping Monitoring Commands

Command	Purpose
Device# show ip igmp ssm-mapping	Displays information about SSM mapping.
Device# show ip igmp ssm-mapping group-address	Displays the sources that SSM mapping uses for a particular group.
Device# show ip igmp groups [<i>group-name</i> <i>group-address</i> <i>interface-type interface-number</i>] [detail]	Displays the multicast groups with receivers that are directly connected to the router and that were learned through IGMP.
Device# show host	Displays the default domain name, the style of name lookup service, a list of name server hosts, and the cached list of hostnames and addresses.
Device# debug ip igmp group-address	Displays the IGMP packets received and sent and IGMP host-related events.

Configuration Examples for SSM and SSM Mapping

SSM with IGMPv3 Example

The following example shows how to configure a device (running IGMPv3) for SSM:

```
ip multicast-routing
!
interface GigabitEthernet3/1/0
 ip address 172.21.200.203 255.255.255.0
 description backbone interface
 ip pim sparse-mode
!
interface GigabitEthernet3/2/0
 ip address 131.108.1.2 255.255.255.0
 ip pim sparse-mode
 description ethernet connected to hosts
 ip igmp version 3
```

```
!
ip pim ssm default
```

SSM Filtering Example

The following example shows how to configure filtering on legacy RP routers running software releases that do not support SSM routing. This filtering will suppress all unwanted PIM-SM and MSDP traffic in the SSM range. Without this filtering, SSM will still operate, but there may be additional RPT traffic if legacy first hop and last hop routers exist in the network.

```
ip access-list extended no-ssm-range
  deny ip any 232.0.0.0 0.255.255.255 ! SSM range
  permit ip any any
! Deny registering in SSM range
ip pim accept-register list no-ssm-range
ip access-list extended msdp-nono-list
  deny ip any 232.0.0.0 0.255.255.255 ! SSM Range
  ! .
  ! .
  ! .
! See ftp://ftpeng.cisco.com/ipmulticast/config-notes/msdp-sa-filter.txt for other SA
! messages that typically need to be filtered.
  permit ip any any
! Filter generated SA messages in SSM range. This configuration is only needed if there
! are directly connected sources to this router. The "ip pim accept-register" command
! filters remote sources.
ip msdp redistribute list msdp-nono-list
! Filter received SA messages in SSM range. "Filtered on receipt" means messages are
! neither processed or forwarded. Needs to be configured for each MSDP peer.
ip msdp sa-filter in msdp-peer1 list msdp-nono-list
! .
! .
! .
ip msdp sa-filter in msdp-peerN list msdp-nono-list
```

SSM Mapping Example

The following configuration example shows a router configuration for SSM mapping. This example also displays a range of other IGMP and SSM configuration options to show compatibility between features. Do not use this configuration example as a model unless you understand all of the features used in the example.



Note Address assignment in the global SSM range 232.0.0.0/8 should be random. If you copy parts or all of this sample configuration, make sure to select a random address range but not 232.1.1.x as shown in this example. Using a random address range minimizes the possibility of address collision and may prevent conflicts when other SSM content is imported while SSM mapping is used.

```
!
no ip domain lookup
ip domain multicast ssm.map.cisco.com
ip name-server 10.48.81.21
!
!
ip multicast-routing distributed
ip igmp ssm-map enable
```

```

ip igmp ssm-map static 10 172.16.8.10
ip igmp ssm-map static 11 172.16.8.11
!
!
.
.
.
!
interface GigabitEthernet0/0/0
description Sample IGMP Interface Configuration for SSM-Mapping Example
ip address 10.20.1.2 255.0.0.0
ip pim sparse-mode
ip igmp last-member-query-interval 100
ip igmp static-group 232.1.2.1 source ssm-map
ip igmp version 3
ip igmp explicit-tracking
ip igmp limit 2
ip igmp v3lite
ip urd
!
.
.
.
!
ip pim ssm default
!
access-list 10 permit 232.1.2.10
access-list 11 permit 232.1.2.0 0.0.0.255
!

```

This table describes the significant commands shown in the SSM mapping configuration example.

Table 34: SSM Mapping Configuration Example Command Descriptions

Command	Description
no ip domain lookup	Disables IP DNS-based hostname-to-address translation. Note The no ip domain-list command is shown in the configuration only to demonstrate that disabling IP DNS-based hostname-to-address translation does not conflict with configuring SSM mapping. If this command is enabled, the Cisco IOS XE software will try to resolve unknown strings as hostnames.
ip domain multicast ssm-map.cisco.com	Specifies ssm-map.cisco.com as the domain prefix for SSM mapping.
ip name-server 10.48.81.21	Specifies 10.48.81.21 as the IP address of the DNS server to be used by SSM mapping and any other service in the software that utilizes DNS.
ip multicast-routing	Enables IP multicast routing.
ip igmp ssm-map enable	Enables SSM mapping.
ip igmp ssm-map static 10 172.16.8.10	Configures the groups permitted by ACL 10 to use source address 172.16.8.10. <ul style="list-style-type: none"> In this example, ACL 10 permits all groups in the 232.1.2.0/25 range except 232.1.2.10.

Command	Description
ip igmp ssm-map static 11 172.16.8.11	Configures the groups permitted by ACL 11 to use source address 172.16.8.11. • In this example, ACL 11 permits group 232.1.2.10.
ip pim sparse-mode	Enables PIM sparse mode.
ip igmp last-member-query-interval 100	Reduces the leave latency for IGMPv2 hosts. Note This command is not required for configuring SSM mapping; however, configuring this command can be beneficial for IGMPv2 hosts relying on SSM mapping.
ip igmp static-group 232.1.2.1 source ssm-map	Configures SSM mapping to be used to determine the sources associated with group 232.1.2.1. The resulting (S, G) channels are statically forwarded.
ip igmp version 3	Enables IGMPv3 on this interface. Note This command is shown in the configuration only to demonstrate that IGMPv3 can be configured simultaneously with SSM mapping; however, it is not required.
ip igmp explicit-tracking	Minimizes the leave latency for IGMPv3 host leaving a multicast channel. Note This command is not required for configuring SSM mapping.
ip igmp limit 2	Limits the number of IGMP states resulting from IGMP membership states on a per-interface basis. Note This command is not required for configuring SSM mapping.
ip igmp v3lite	Enables the acceptance and processing of IGMP v3lite membership reports on this interface. Note This command is shown in the configuration only to demonstrate that IGMP v3lite can be configured simultaneously with SSM mapping; however, it is not required.
ip urd	Enables interception of TCP packets sent to the reserved URD port 465 on an interface and processing of URD channel subscription reports. Note This command is shown in the configuration only to demonstrate that URD can be configured simultaneously with SSM mapping; however, it is not required.
ip pim ssm default	Configures SSM service. The default keyword defines the SSM range access list as 232/8.

Command	Description
<code>access-list 10 permit 232.1.2.10</code> <code>access-list 11 permit 232.1.2.0</code> <code>0.0.0.255</code>	Configures the ACLs to be used for static SSM mapping. Note These are the ACLs that are referenced by the <code>ip igmp ssm-map static</code> commands in this configuration example.

DNS Server Configuration Example

To configure DNS-based SSM mapping, you need to create a DNS server zone or add records to an existing zone. If the routers that are using DNS-based SSM mapping are also using DNS for other purposes besides SSM mapping, you should use a normally-configured DNS server. If DNS-based SSM mapping is the only DNS implementation being used on the router, you can configure a fake DNS setup with an empty root zone, or a root zone that points back to itself.

The following example shows how to create a zone and import the zone data using Network Registrar:

```
Router> zone 1.1.232.ssm-map.cisco.com. create primary file=named.ssm-map
100 Ok
Router> dns reload
100 Ok
```

The following example shows how to import the zone files from a named.conf file for BIND 8:

```
Router> ::import named.conf /etc/named.conf
Router> dns reload
100 Ok:
```



Note Network Registrar version 8.0 and later support import BIND 8 format definitions.

Additional References

Related Documents

Related Topic	Document Title
For complete syntax and usage information for the commands used in this chapter.	<i>IP Multicast Command Reference, Cisco IOS Release 15.2(2)E (Catalyst 2960-XR Switch) Command Reference (Catalyst 9500 Series Switches) Command Reference (Catalyst 9300 Series Switches)</i>

Error Message Decoder

Description	Link
To help you research and resolve system error messages in this release, use the Error Message Decoder tool.	https://www.cisco.com/cgi-bin/Support/Errordecoder/index.cgi

Standards and RFCs

Standard/RFC	Title
RFC 4601	<i>Protocol-Independent Multicast-Sparse Mode (PIM-SM): Protocol Specification</i>

MIBs

MIB	MIBs Link
All the supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
<p>The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies.</p> <p>To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds.</p> <p>Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.</p>	http://www.cisco.com/support

Feature History and Information for SSM

Release	Modification
Cisco IOS Release 15.0(2)EX1	This feature was introduced.



PART **III**

First Hop Protocols

- [Configuring HSRP and VRRP, on page 283](#)



CHAPTER 20

Configuring HSRP and VRRP

- [Configuring HSRP and VRRP, on page 283](#)

Configuring HSRP and VRRP

This chapter describes how to use Hot Standby Router Protocol (HSRP) to provide routing redundancy for routing IP traffic without being dependent on the availability of any single router.

You can also use a version of HSRP in Layer 2 mode to configure a redundant command switch to take over cluster management if the cluster command switch fails.

Virtual Router Redundancy Protocol (VRRP) is supported for IPv4 on switches running IP Lite image.

Information About Configuring HSRP

HSRP Overview

HSRP is Cisco's standard method of providing high network availability by providing first-hop redundancy for IP hosts on an IEEE 802 LAN configured with a default gateway IP address. HSRP routes IP traffic without relying on the availability of any single router. It enables a set of router interfaces to work together to present the appearance of a single virtual router or default gateway to the hosts on a LAN. When HSRP is configured on a network or segment, it provides a virtual Media Access Control (MAC) address and an IP address that is shared among a group of configured routers. HSRP allows two or more HSRP-configured routers to use the MAC address and IP network address of a virtual router. The virtual router does not exist; it represents the common target for routers that are configured to provide backup to each other. One of the routers is selected to be the active router and another to be the standby router, which assumes control of the group MAC address and IP address should the designated active router fail.



Note Routers in an HSRP group can be any router interface that supports HSRP, including routed ports and switch virtual interfaces (SVIs).

HSRP provides high network availability by providing redundancy for IP traffic from hosts on networks. In a group of router interfaces, the active router is the router of choice for routing packets; the standby router is the router that takes over the routing duties when an active router fails or when preset conditions are met.

HSRP is useful for hosts that do not support a router discovery protocol and cannot switch to a new router when their selected router reloads or loses power. When HSRP is configured on a network segment, it provides a virtual MAC address and an IP address that is shared among router interfaces in a group of router interfaces running HSRP. The router selected by the protocol to be the active router receives and routes packets destined for the group's MAC address. For n routers running HSRP, there are $n + 1$ IP and MAC addresses assigned.

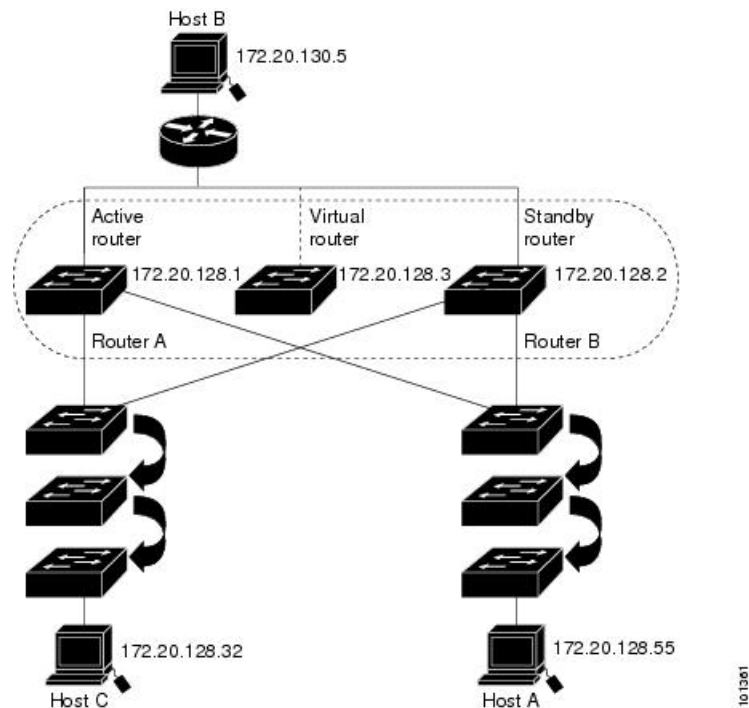
HSRP detects when the designated active router fails, and a selected standby router assumes control of the Hot Standby group's MAC and IP addresses. A new standby router is also selected at that time. Devices running HSRP send and receive multicast UDP-based hello packets to detect router failure and to designate active and standby routers. When HSRP is configured on an interface, Internet Control Message Protocol (ICMP) redirect messages are automatically enabled for the interface.

You can configure multiple Hot Standby groups among switches and switch stacks that are operating in Layer 3 to make more use of the redundant routers.

To do so, specify a group number for each Hot Standby command group you configure for an interface. For example, you might configure an interface on switch 1 as an active router and one on switch 2 as a standby router and also configure another interface on switch 2 as an active router with another interface on switch 1 as its standby router.

The following figure shows a segment of a network configured for HSRP. Each router is configured with the MAC address and IP network address of the virtual router. Instead of configuring hosts on the network with the IP address of Router A, you configure them with the IP address of the virtual router as their default router. When Host C sends packets to Host B, it sends them to the MAC address of the virtual router. If for any reason, Router A stops transferring packets, Router B responds to the virtual IP address and virtual MAC address and becomes the active router, assuming the active router duties. Host C continues to use the IP address of the virtual router to address packets destined for Host B, which Router B now receives and sends to Host B. Until Router A resumes operation, HSRP allows Router B to provide uninterrupted service to users on Host C's segment that need to communicate with users on Host B's segment and also continues to perform its normal function of handling packets between the Host A segment and Host B.

Figure 16: Typical HSRP Configuration



HSRP Versions

and later support these Hot Standby Router Protocol (HSRP) versions:

The switch supports these HSRP versions:

- HSRPv1- Version 1 of the HSRP, the default version of HSRP. It has these features:
 - The HSRP group number can be from 0 to 255.
 - HSRPv1 uses the multicast address 224.0.0.2 to send hello packets, which can conflict with Cisco Group Management Protocol (CGMP) leave processing. You cannot enable HSRPv1 and CGMP at the same time; they are mutually exclusive.
- HSRPv2- Version 2 of the HSRP has these features:
 - To match the HSRP group number to the VLAN ID of a subinterface, HSRPv2 can use a groupnumber from 0 to 4095 and a MAC address from 0000.0C9F.F000 to 0000.0C9F.FFFF.
 - HSRPv2 uses the multicast address 224.0.0.102 to send hello packets. HSRPv2 and CGMP leave processing are no longer mutually exclusive, and both can be enabled at the same time.
 - HSRPv2 has a different packet format than HRSPv1.

A switch running HSRPv1 cannot identify the physical router that sent a hello packet because the source MAC address of the router is the virtual MAC address.

HSRPv2 has a different packet format than HSRPv1. A HSRPv2 packet uses the type-length-value (TLV) format and has a 6-byte identifier field with the MAC address of the physical router that sent the packet.

If an interface running HSRPv1 gets an HSRPv2 packet, the type field is ignored.

Multiple HSRP

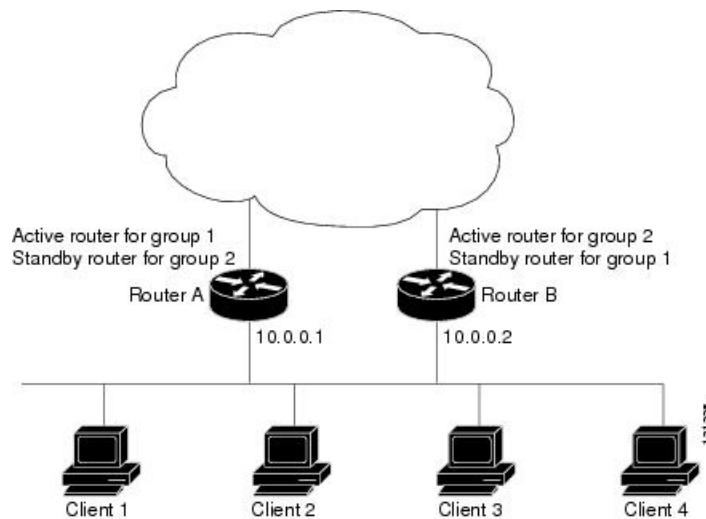
The switch supports Multiple HSRP (MHSRP), an extension of HSRP that allows load sharing between two or more HSRP groups. You can configure MHSRP to achieve load-balancing and to use two or more standby groups (and paths) from a host network to a server network.

In the figure below, half the clients are configured for Router A, and half the clients are configured for Router B. Together, the configuration for Routers A and B establishes two HSRP groups. For group 1, Router A is the default active router because it has the assigned highest priority, and Router B is the standby router. For group 2, Router B is the default active router because it has the assigned highest priority, and Router A is the standby router. During normal operation, the two routers share the IP traffic load. When either router becomes unavailable, the other router becomes active and assumes the packet-transfer functions of the router that is unavailable.



Note For MHSRP, you need to enter the **standby preempt** interface configuration command on the HSRP interfaces so that if a router fails and then comes back up, preemption restores load sharing.

Figure 17: MHSRP Load Sharing



SSO HSRP

SSO HSRP alters the behavior of HSRP when a device with redundant Route Processors (RPs) is configured for stateful switchover (SSO) redundancy mode. When an RP is active and the other RP is standby, SSO enables the standby RP to take over if the active RP fails.

With this functionality, HSRP SSO information is synchronized to the standby RP, allowing traffic that is sent using the HSRP virtual IP address to be continuously forwarded during a switchover without a loss of data or a path change. Additionally, if both RPs fail on the active HSRP device, then the standby HSRP device takes over as the active HSRP device.

The feature is enabled by default when the redundancy mode of operation is set to SSO.

HSRP and Switch Stacks

HSRP hello messages are generated by the active switch. If HSRP fails on the active switch, a flap in the HSRP active state might occur. This is because HSRP hello messages are not generated while a new active switch is elected and initialized, and the standby switch might become active after the active switch fails.

How to Configure HSRP

Default HSRP Configuration

Table 35: Default HSRP Configuration

Feature	Default Setting
HSRP version	Version 1
HSRP groups	None configured
Standby group number	0
Standby MAC address	System assigned as: 0000.0c07.acXX, where XX is the HSRP group number
Standby priority	100
Standby delay	0 (no delay)
Standby track interface priority	10
Standby hello time	3 seconds
Standby holdtime	10 seconds

HSRP Configuration Guidelines

- HSRPv2 and HSRPv1 are mutually exclusive. HSRPv2 is not interoperable with HSRPv1 on an interface and the reverse.
- In the procedures, the specified interface must be one of these Layer 3 interfaces:
 - Routed port: A physical port configured as a Layer 3 port by entering the **no switchport** command in interface configuration mode.
 - SVI: A VLAN interface created by using the **interface vlan** *vlan_id* in global configuration mode, and by default a Layer 3 interface.
 - Etherchannel port channel in Layer 3 mode: A port-channel logical interface created by using the **interface port-channel** *port-channel-number* in global configuration mode, and binding the Ethernet interface into the channel group.
- All Layer 3 interfaces must have IP addresses assigned to them.
- The version of an HSRP group can be changed from HSRPv2 to HSRPv1 only if the group number is less than 256.

- If you change the HSRP version on an interface, each HSRP group resets because it now has a new virtual MAC address.
- Only on mixed stacks of Catalyst switches:
- HSRP groups can be configured up to 32 instances.
- Configure only one instance of a First Hop Redundancy Protocol (FHRP). The switches support HSRPv1, HSRPv2, and HSRP for IPv6.
- HSRP for IPv4 and HSRP for IPv6 are mutually exclusive. You cannot enable both at the same time.
- When configuring group numbers for HSRPv2 and HSRP, you must use group numbers in ranges that are multiples of 256. Valid ranges are 0 to 255, 256 to 511, 512 to 767, 3840 to 4095, and so on.
- Examples of valid and invalid group numbers:
- If you configure groups with the numbers 2, 150, and 225, you cannot configure another group with the number 3850. It is not in the range of 0 to 255.
- If you configure groups with the numbers 520, 600, and 700, you cannot configure another group with the number 900. It is not in the range of 512 to 767.

Enabling HSRP

The **standby ip** interface configuration command activates HSRP on the configured interface. If an IP address is specified, that address is used as the designated address for the Hot Standby group. If no IP address is specified, the address is learned through the standby function. You must configure at least one Layer 3 port on the LAN with the designated address. Configuring an IP address always overrides another designated address currently in use.

When the **standby ip** command is enabled on an interface and proxy ARP is enabled, if the interface's Hot Standby state is active, proxy ARP requests are answered using the Hot Standby group MAC address. If the interface is in a different state, proxy ARP responses are suppressed.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **standby version** { **1** | **2** }
4. **standby** [*group-number*] **ip** [*ip-address*] [**secondary**]
5. **end**
6. **show standby** [*interface-id*] [*group*]
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Switch(config)# configure terminal	Enters global configuration mode.

	Command or Action	Purpose
Step 2	interface <i>interface-id</i> Example: Switch(config)# interface gigabitethernet1/0/1	Enters interface configuration mode, and enter the Layer 3 interface on which you want to enable HSRP.
Step 3	standby version { 1 2 } Example: Switch(config-if)# standby version 1	(Optional) Configures the HSRP version on the interface. <ul style="list-style-type: none"> • 1- Selects HSRPv1. • 2- Selects HSRPv2. If you do not enter this command or do not specify a keyword, the interface runs the default HSRP version, HSRP v1.
Step 4	standby [<i>group-number</i>] ip [<i>ip-address</i> [<i>secondary</i>]] Example: Switch(config-if)# standby 1 ip	Creates (or enable) the HSRP group using its number and virtual IP address. <ul style="list-style-type: none"> • (Optional) <i>group-number</i>- The group number on the interface for which HSRP is being enabled. The range is 0 to 255; the default is 0. If there is only one HSRP group, you do not need to enter a group number. • (Optional on all but one interface) <i>ip-address</i>- The virtual IP address of the hot standby router interface. You must enter the virtual IP address for at least one of the interfaces; it can be learned on the other interfaces. • (Optional) secondary- The IP address is a secondary hot standby router interface. If neither router is designated as a secondary or standby router and no priorities are set, the primary IP addresses are compared and the higher IP address is the active router, with the next highest as the standby router.
Step 5	end Example: Switch(config-if)# end	Returns to privileged EXEC mode
Step 6	show standby [<i>interface-id</i> [<i>group</i>]] Example: Switch # show standby	Verifies the configuration of the standby groups.
Step 7	copy running-config startup-config Example: Switch# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Configuring HSRP Priority

The **standby priority**, **standby preempt**, and **standby track** interface configuration commands are all used to set characteristics for finding active and standby routers and behavior regarding when a new active router takes over.

When configuring HSRP priority, follow these guidelines:

- Assigning a priority allows you to select the active and standby routers. If preemption is enabled, the router with the highest priority becomes the active router. If priorities are equal, the current active router does not change.
- The highest number (1 to 255) represents the highest priority (most likely to become the active router).
- When setting the priority, preempt, or both, you must specify at least one keyword (**priority**, **preempt**, or both)
- The priority of the device can change dynamically if an interface is configured with the **standby track** command and another interface on the router goes down.
- The **standby track** interface configuration command ties the router hot standby priority to the availability of its interfaces and is useful for tracking interfaces that are not configured for HSRP. When a tracked interface fails, the hot standby priority on the device on which tracking has been configured decreases by 10. If an interface is not tracked, its state changes do not affect the hot standby priority of the configured device. For each interface configured for hot standby, you can configure a separate list of interfaces to be tracked
- The **standby track interface-priority** interface configuration command specifies how much to decrement the hot standby priority when a tracked interface goes down. When the interface comes back up, the priority is incremented by the same amount.
- When multiple tracked interfaces are down and *interface-priority* values have been configured, the configured priority decrements are cumulative. If tracked interfaces that were not configured with priority values fail, the default decrement is 10, and it is noncumulative.
- When routing is first enabled for the interface, it does not have a complete routing table. If it is configured to preempt, it becomes the active router, even though it is unable to provide adequate routing services. To solve this problem, configure a delay time to allow the router to update its routing table.

Beginning in privileged EXEC mode, use one or more of these steps to configure HSRP priority characteristics on an interface:

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **standby** [*group-number*] **priority***priority*
4. **standby** [*group-number*] **preempt** [**delay** [*minimumseconds*] [**reload***seconds*] [**sync***seconds*]]
5. **standby** [*group-number*] **track** *type number* [*interface-priority*]
6. **end**
7. **show running-config**
8. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Switch # configure terminal	Enters global configuration mode.
Step 2	interface interface-id Example: Switch(config)# interface gigabitethernet1/0/1	Enters interface configuration mode, and enter the HSRP interface on which you want to set priority.
Step 3	standby [group-number] prioritypriority Example: Switch(config-if)# standby 120 priority 50	Sets a priority value used in choosing the active router. The range is 1 to 255; the default priority is 100. The highest number represents the highest priority. <ul style="list-style-type: none"> (Optional) group-number—The group number to which the command applies. Use the no form of the command to restore the default values.
Step 4	standby [group-number] preempt [delay [minimumseconds] [reloadseconds] [syncseconds]] Example: Switch(config-if)# standby 1 preempt delay 300	Configures the router to preempt , which means that when the local router has a higher priority than the active router, it becomes the active router. <ul style="list-style-type: none"> (Optional) group-number—The group number to which the command applies. (Optional) delay minimum—Set to cause the local router to postpone taking over the active role for the number of seconds shown. The range is 0 to 3600 seconds (1 hour); the default is 0 (no delay before taking over). (Optional) delay reload—Set to cause the local router to postpone taking over the active role after a reload for the number of seconds shown. The range is 0 to 3600 seconds (1 hour); the default is 0 (no delay before taking over after a reload). (Optional) delay sync—Set to cause the local router to postpone taking over the active role so that IP redundancy clients can reply (either with an ok or wait reply) for the number of seconds shown. The range is 0 to 3600 seconds (1 hour); the default is 0 (no delay before taking over). Use the no form of the command to restore the default values.
Step 5	standby [group-number] track type number [interface-priority] Example: Switch(config-if)# standby track interface gigabitethernet1/1/1	Configures an interface to track other interfaces so that if one of the other interfaces goes down, the device's Hot Standby priority is lowered. <ul style="list-style-type: none"> (Optional) group-number—The group number to which the command applies.

	Command or Action	Purpose
		<ul style="list-style-type: none"> • type- Enter the interface type (combined with interface number) that is tracked. • number- Enter the interface number (combined with interface type) that is tracked. • (Optional) interface-priority- Enter the amount by which the hot standby priority for the router is decremented or incremented when the interface goes down or comes back up. The default value is 10.
Step 6	end Example: Switch(config-if) # end	Returns to privileged EXEC mode.
Step 7	show running-config	Verifies the configuration of the standby groups.
Step 8	copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Configuring MHSRP

To enable MHSRP and load-balancing, you configure two routers as active routers for their groups, with virtual routers as standby routers as shown in the *MHSRP Load Sharing* figure in the Multiple HSRP section. You need to enter the **standby preempt** interface configuration command on each HSRP interface so that if a router fails and comes back up, the preemption occurs and restores load-balancing.

Router A is configured as the active router for group 1, and Router B is configured as the active router for group 2. The HSRP interface for Router A has an IP address of 10.0.0.1 with a group 1 standby priority of 110 (the default is 100). The HSRP interface for Router B has an IP address of 10.0.0.2 with a group 2 standby priority of 110.

Group 1 uses a virtual IP address of 10.0.0.3 and group 2 uses a virtual IP address of 10.0.0.4.

Configuring Router A

SUMMARY STEPS

1. **configure terminal**
2. **interface** *type number*
3. **no switchport**
4. **ip address** *ip-address mask*
5. **standby** [*group-number*] **ip** [*ip-address*] [**secondary**]
6. **standby** [*group-number*] **priority** *priority*
7. **standby** [*group-number*] **preempt** [**delay** [*minimum seconds*]] [**reload** *seconds*] [**sync** *seconds*]
8. **standby** [*group-number*] **ip** [*ip-address*] [**secondary**]
9. **standby** [*group-number*] **preempt** [**delay** [*minimum seconds*]] [**reload** *seconds*] [**sync** *seconds*]
10. **end**
11. **show running-config**
12. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Switch # configure terminal	Enters global configuration mode.
Step 2	interface <i>type number</i> Example: Switch (config)# interface <code>gigabitethernet1/0/1</code>	Configures an interface type and enters interface configuration mode.
Step 3	no switchport Example: Switch (config)# no switchport	Switches an interface that is in Layer 2 mode into Layer 3 mode for Layer 3 configuration.
Step 4	ip address <i>ip-address mask</i> Example: Switch (config-if)# ip address <code>10.0.0.1 255.255.255.0</code>	Specifies an IP address for an interface.
Step 5	standby [<i>group-number</i>] ip [<i>ip-address</i> [secondary]] Example: Switch (config-if)# standby 1 ip 10.0.0.3	<p>Creates the HSRP group using its number and virtual IP address.</p> <ul style="list-style-type: none"> • (Optional) <i>group-number</i>- The group number on the interface for which HSRP is being enabled. The range is 0 to 255; the default is 0. If there is only one HSRP group, you do not need to enter a group number. • (Optional on all but one interface) <i>ip-address</i>- The virtual IP address of the hot standby router interface. You must enter the virtual IP address for at least one of the interfaces; it can be learned on the other interfaces. • (Optional) secondary- The IP address is a secondary hot standby router interface. If neither router is designated as a secondary or standby router and no priorities are set, the primary IP addresses are compared and the higher IP address is the active router, with the next highest as the standby router.
Step 6	standby [<i>group-number</i>] priority <i>priority</i> Example: Switch(config-if)# standby 1 priority 110	<p>Sets a priority value used in choosing the active router. The range is 1 to 255; the default priority is 100. The highest number represents the highest priority.</p> <ul style="list-style-type: none"> • (Optional) <i>group-number</i>—The group number to which the command applies. <p>Use the no form of the command to restore the default values.</p>

	Command or Action	Purpose
<p>Step 7</p>	<p>standby [<i>group-number</i>] preempt [delay [<i>minimum seconds</i>] [reload <i>seconds</i>] [sync <i>seconds</i>]]</p> <p>Example:</p> <pre>Switch(config-if)# standby 1 preempt delay 300</pre>	<p>Configures the router to preempt, which means that when the local router has a higher priority than the active router, it becomes the active router.</p> <ul style="list-style-type: none"> • (Optional) group-number—The group number to which the command applies. • (Optional) delay minimum—Set to cause the local router to postpone taking over the active role for the number of seconds shown. The range is 0 to 3600 seconds (1 hour); the default is 0 (no delay before taking over). • (Optional) delay reload—Set to cause the local router to postpone taking over the active role after a reload for the number of seconds shown. The range is 0 to 3600 seconds (1 hour); the default is 0 (no delay before taking over after a reload). • (Optional) delay sync—Set to cause the local router to postpone taking over the active role so that IP redundancy clients can reply (either with an ok or wait reply) for the number of seconds shown. The range is 0 to 3600 seconds (1 hour); the default is 0 (no delay before taking over). <p>Use the no form of the command to restore the default values.</p>
<p>Step 8</p>	<p>standby [<i>group-number</i>] ip [<i>ip-address</i> [secondary]]</p> <p>Example:</p> <pre>Switch (config-if)# standby 2 ip 10.0.0.4</pre>	<p>Creates the HSRP group using its number and virtual IP address.</p> <ul style="list-style-type: none"> • (Optional) <i>group-number</i>— The group number on the interface for which HSRP is being enabled. The range is 0 to 255; the default is 0. If there is only one HSRP group, you do not need to enter a group number. • (Optional on all but one interface) <i>ip-address</i>— The virtual IP address of the hot standby router interface. You must enter the virtual IP address for at least one of the interfaces; it can be learned on the other interfaces. • (Optional) secondary— The IP address is a secondary hot standby router interface. If neither router is designated as a secondary or standby router and no priorities are set, the primary IP addresses are compared and the higher IP address is the active router, with the next highest as the standby router.
<p>Step 9</p>	<p>standby [<i>group-number</i>] preempt [delay [<i>minimum seconds</i>] [reload <i>seconds</i>] [sync <i>seconds</i>]]</p> <p>Example:</p> <pre>Switch(config-if)# standby 2 preempt delay 300</pre>	<p>Configures the router to preempt, which means that when the local router has a higher priority than the active router, it becomes the active router.</p> <ul style="list-style-type: none"> • (Optional) group-number—The group number to which the command applies.

	Command or Action	Purpose
		<ul style="list-style-type: none"> • (Optional) delay minimum—Set to cause the local router to postpone taking over the active role for the number of seconds shown. The range is 0 to 3600 seconds (1 hour); the default is 0 (no delay before taking over). • (Optional) delay reload—Set to cause the local router to postpone taking over the active role after a reload for the number of seconds shown. The range is 0 to 3600 seconds (1 hour); the default is 0 (no delay before taking over after a reload). • (Optional) delay sync—Set to cause the local router to postpone taking over the active role so that IP redundancy clients can reply (either with an ok or wait reply) for the number of seconds shown. The range is 0 to 3600 seconds (1 hour); the default is 0 (no delay before taking over). <p>Use the no form of the command to restore the default values.</p>
Step 10	end Example: <code>Switch(config-if)# end</code>	Returns to privileged EXEC mode.
Step 11	show running-config	Verifies the configuration of the standby groups.
Step 12	copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Configuring Router B

SUMMARY STEPS

1. **configure terminal**
2. **interface** *type number*
3. **no switchport**
4. **ip address** *ip-address mask*
5. **standby** [*group-number*] **ip** [*ip-address* [**secondary**]]
6. **standby** [*group-number*] **priority** *priority*
7. **standby** [*group-number*] **preempt** [**delay** [*minimum seconds*] [**reload** *seconds*] [**sync** *seconds*]]
8. **standby** [*group-number*] **ip** [*ip-address* [**secondary**]]
9. **standby** [*group-number*] **preempt** [**delay** [*minimum seconds*] [**reload** *seconds*] [**sync** *seconds*]]
10. **end**
11. **show running-config**
12. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Switch # configure terminal	Enters global configuration mode.
Step 2	interface type number Example: Switch (config)# interface gigabitethernet1/0/1	Configures an interface type and enters interface configuration mode.
Step 3	no switchport Example: Switch (config)# no switchport	Switches an interface that is in Layer 2 mode into Layer 3 mode for Layer 3 configuration.
Step 4	ip address ip-address mask Example: Switch (config-if)# 10.0.0.2 255.255.255.0	Specifies an IP address for an interface.
Step 5	standby [group-number] ip [ip-address [secondary]] Example: Switch (config-if)# standby 1 ip 10.0.0.3	Creates the HSRP group using its number and virtual IP address. <ul style="list-style-type: none"> • (Optional) <i>group-number</i>- The group number on the interface for which HSRP is being enabled. The range is 0 to 255; the default is 0. If there is only one HSRP group, you do not need to enter a group number. • (Optional on all but one interface) <i>ip-address</i>- The virtual IP address of the hot standby router interface. You must enter the virtual IP address for at least one of the interfaces; it can be learned on the other interfaces. • (Optional) secondary- The IP address is a secondary hot standby router interface. If neither router is designated as a secondary or standby router and no priorities are set, the primary IP addresses are compared and the higher IP address is the active router, with the next highest as the standby router.
Step 6	standby [group-number] priority priority Example: Switch(config-if)# standby 1 priority 110	Sets a priority value used in choosing the active router. The range is 1 to 255; the default priority is 100. The highest number represents the highest priority. <ul style="list-style-type: none"> • (Optional) <i>group-number</i>—The group number to which the command applies. <p>Use the no form of the command to restore the default values.</p>

	Command or Action	Purpose
Step 7	<p>standby [<i>group-number</i>] preempt [delay [<i>minimum seconds</i>] [reload <i>seconds</i>] [sync <i>seconds</i>]]</p> <p>Example:</p> <pre>Switch(config-if)# standby 1 preempt delay 300</pre>	<p>Configures the router to preempt, which means that when the local router has a higher priority than the active router, it becomes the active router.</p> <ul style="list-style-type: none"> • (Optional) group-number—The group number to which the command applies. • (Optional) delay minimum—Set to cause the local router to postpone taking over the active role for the number of seconds shown. The range is 0 to 3600 seconds (1 hour); the default is 0 (no delay before taking over). • (Optional) delay reload—Set to cause the local router to postpone taking over the active role after a reload for the number of seconds shown. The range is 0 to 3600 seconds (1 hour); the default is 0 (no delay before taking over after a reload). • (Optional) delay sync—Set to cause the local router to postpone taking over the active role so that IP redundancy clients can reply (either with an ok or wait reply) for the number of seconds shown. The range is 0 to 3600 seconds (1 hour); the default is 0 (no delay before taking over). <p>Use the no form of the command to restore the default values.</p>
Step 8	<p>standby [<i>group-number</i>] ip [<i>ip-address</i> [secondary]]</p> <p>Example:</p> <pre>Switch (config-if)# standby 2 ip 10.0.0.4</pre>	<p>Creates the HSRP group using its number and virtual IP address.</p> <ul style="list-style-type: none"> • (Optional) <i>group-number</i>— The group number on the interface for which HSRP is being enabled. The range is 0 to 255; the default is 0. If there is only one HSRP group, you do not need to enter a group number. • (Optional on all but one interface) <i>ip-address</i>— The virtual IP address of the hot standby router interface. You must enter the virtual IP address for at least one of the interfaces; it can be learned on the other interfaces. • (Optional) secondary— The IP address is a secondary hot standby router interface. If neither router is designated as a secondary or standby router and no priorities are set, the primary IP addresses are compared and the higher IP address is the active router, with the next highest as the standby router.
Step 9	<p>standby [<i>group-number</i>] preempt [delay [<i>minimum seconds</i>] [reload <i>seconds</i>] [sync <i>seconds</i>]]</p> <p>Example:</p> <pre>Switch(config-if)# standby 2 preempt delay 300</pre>	<p>Configures the router to preempt, which means that when the local router has a higher priority than the active router, it becomes the active router.</p> <ul style="list-style-type: none"> • (Optional) group-number—The group number to which the command applies.

	Command or Action	Purpose
		<ul style="list-style-type: none"> • (Optional) delay minimum—Set to cause the local router to postpone taking over the active role for the number of seconds shown. The range is 0 to 3600 seconds (1 hour); the default is 0 (no delay before taking over). • (Optional) delay reload—Set to cause the local router to postpone taking over the active role after a reload for the number of seconds shown. The range is 0 to 3600 seconds (1 hour); the default is 0 (no delay before taking over after a reload). • (Optional) delay sync—Set to cause the local router to postpone taking over the active role so that IP redundancy clients can reply (either with an ok or wait reply) for the number of seconds shown. The range is 0 to 3600 seconds (1 hour); the default is 0 (no delay before taking over). <p>Use the no form of the command to restore the default values.</p>
Step 10	end Example: <code>Switch(config-if)# end</code>	Returns to privileged EXEC mode.
Step 11	show running-config	Verifies the configuration of the standby groups.
Step 12	copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Configuring HSRP Authentication and Timers

You can optionally configure an HSRP authentication string or change the hello-time interval and holdtime.

When configuring these attributes, follow these guidelines:

- The authentication string is sent unencrypted in all HSRP messages. You must configure the same authentication string on all routers and access servers on a cable to ensure interoperation. Authentication mismatch prevents a device from learning the designated Hot Standby IP address and timer values from other routers configured with HSRP.
- Routers or access servers on which standby timer values are not configured can learn timer values from the active or standby router. The timers configured on an active router always override any other timer settings.
- All routers in a Hot Standby group should use the same timer values. Normally, the *holdtime* is greater than or equal to 3 times the *hellotime*.

Beginning in privileged EXEC mode, use one or more of these steps to configure HSRP authentication and timers on an interface:

SUMMARY STEPS

1. configure terminal

2. **interface** *interface-id*
3. **standby** [*group-number*] **authentication** *string*
4. **standby** [*group-number*] **timers** *hellotime holdtime*
5. **end**
6. **show running-config**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Switch # configure terminal	Enters global configuration mode.
Step 2	interface <i>interface-id</i> Example: Switch(config) # interface gigabitethernet1/0/1	Enters interface configuration mode, and enter the HSRP interface on which you want to set priority.
Step 3	standby [<i>group-number</i>] authentication <i>string</i> Example: Switch(config-if) # standby 1 authentication word	(Optional) authentication <i>string</i> —Enter a string to be carried in all HSRP messages. The authentication string can be up to eight characters in length; the default string is cisco . (Optional) <i>group-number</i> —The group number to which the command applies.
Step 4	standby [<i>group-number</i>] timers <i>hellotime holdtime</i> Example: Switch(config-if) # standby 1 timers 5 15	(Optional) Configure the time between hello packets and the time before other routers declare the active router to be down. <ul style="list-style-type: none"> • <i>group-number</i>—The group number to which the command applies. • <i>hellotime</i> —Set the interval between successive hello packets in seconds. The range is 1 to 255 seconds. The default is 3. • <i>holdtime</i>—Set to cause the local router to postpone taking over the active role after a reload for the number of seconds shown. The range is 0 to 3600 seconds (1 hour). The default is 0 (no delay before taking over after a reload).
Step 5	end Example: Switch(config-if) # end	Returns to privileged EXEC mode.
Step 6	show running-config	Verifies the configuration of the standby groups.
Step 7	copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Enabling HSRP Support for ICMP Redirect Messages

ICMP redirect messages are automatically enabled on interfaces configured with HSRP. ICMP is a network layer Internet protocol that provides message packets to report errors and other information relevant to IP processing. ICMP provides diagnostic functions, such as sending and directing error packets to the host. This feature filters outgoing ICMP redirect messages through HSRP, in which the next hop IP address might be changed to an HSRP virtual IP address. For more information, see the Cisco IOS IP Configuration Guide, Release 12.4.

Configuring HSRP Groups and Clustering

When a device is participating in an HSRP standby routing and clustering is enabled, you can use the same standby group for command switch redundancy and HSRP redundancy. Use the **cluster standby-group** *HSRP-group-name* [**routing-redundancy**] global configuration command to enable the same HSRP standby group to be used for command switch and routing redundancy. If you create a cluster with the same HSRP standby group name without entering the **routing-redundancy** keyword, HSRP standby routing is disabled for the group.

Troubleshooting HSRP

If one of the situations as shown in the following table occurs, this message appears:

```
%FHRP group not consistent with already configured groups on the switch stack - virtual MAC reservation failed
```

Table 36: Troubleshooting HSRP

Situation	Action
You configure more than 32 HSRP group instances.	Remove HSRP groups so that up to 32 group instances are configured.
You configure HSRP for IPv4 and HSRP for IPv6 at the same time	Configure either HSRP for IPv4 or HSRP for IPv6 on the switch.
You configure group numbers that are not in valid ranges of 256.	Configure group numbers in a valid range.

Verifying HSRP

Verifying HSRP Configurations

From privileged EXEC mode, use this command to display HSRP settings:

```
show standby [interface-id [group]] [brief] [detail]
```

You can display HSRP information for the whole switch, for a specific interface, for an HSRP group, or for an HSRP group on an interface. You can also specify whether to display a concise overview of HSRP information or detailed HSRP information. The default display is **detail**. If there are a large number of HSRP groups, using the **show standby** command without qualifiers can result in an unwieldy display.

Example

```
Switch #show standby
VLAN1 - Group 1
Local state is Standby, priority 105, may preempt
Hellotime 3 holdtime 10
Next hello sent in 00:00:02.182
Hot standby IP address is 172.20.128.3 configured
Active router is 172.20.128.1 expires in 00:00:09
Standby router is local
Standby virtual mac address is 0000.0c07.ac01
Name is bbb

VLAN1 - Group 100
Local state is Standby, priority 105, may preempt
Hellotime 3 holdtime 10
Next hello sent in 00:00:02.262
Hot standby IP address is 172.20.138.51 configured
Active router is 172.20.128.1 expires in 00:00:09
Active router is local
Standby router is unknown expired
Standby virtual mac address is 0000.0c07.ac64
Name is test
```

Configuration Examples for Configuring HSRP

Enabling HSRP: Example

This example shows how to activate HSRP for group 1 on an interface. The IP address used by the hot standby group is learned by using HSRP.



Note This procedure is the minimum number of steps required to enable HSRP. Other configurations are optional.

```
Switch # configure terminal
Switch(config) # interface gigabitethernet1/0/1
Switch(config-if) # no switchport
Switch(config-if) # standby 1 ip
Switch(config-if) # end
Switch # show standby
```

Configuring HSRP Priority: Example

This example activates a port, sets an IP address and a priority of 120 (higher than the default value), and waits for 300 seconds (5 minutes) before attempting to become the active router:

```
Switch # configure terminal
Switch(config) # interface gigabitethernet1/0/1
Switch(config-if) # no switchport
Switch(config-if) # standby ip 172.20.128.3
Switch(config-if) # standby priority 120 preempt delay 300
Switch(config-if) # end
Switch # show standby
```

Configuring MHSRP: Example

This example shows how to enable the MHSRP configuration shown in the figure *MHSRP Load Sharing*

Router A Configuration

```
Switch # configure terminal
Switch(config) # interface gigabitethernet1/0/1
Switch(config-if) # no switchport
Switch(config-if) # ip address 10.0.0.1 255.255.255.0
Switch(config-if) # standby ip 10.0.0.3
Switch(config-if) # standby 1 priority 110
Switch(config-if) # standby 1 preempt
Switch(config-if) # standby 2 ip 10.0.0.4
Switch(config-if) # standby 2 preempt
Switch(config-if) # end
```

Router B Configuration

```
Switch # configure terminal
Switch(config) # interface gigabitethernet1/0/1
Switch(config-if) # no switchport
Switch(config-if) # ip address 10.0.0.2 255.255.255.0
Switch(config-if) # standby ip 10.0.0.3
Switch(config-if) # standby 1 preempt
Switch(config-if) # standby 2 ip 10.0.0.4
Switch(config-if) # standby 1 priority 110
Switch(config-if) # standby 2 preempt
Switch(config-if) # end
```

Configuring HSRP Authentication and Timer: Example

This example shows how to configure word as the authentication string required to allow Hot Standby routers in group 1 to interoperate:

```
Switch # configure terminal
Switch(config) # interface gigabitethernet1/0/1
Switch(config-if) # no switchport
Switch(config-if) # standby 1 authentication word
Switch(config-if) # end
```

This example shows how to set the timers on standby group 1 with the time between hello packets at 5 seconds and the time after which a router is considered down to be 15 seconds:

```
Switch # configure terminal
Switch(config) # interface gigabitethernet1/0/1
Switch(config-if) # no switchport
Switch(config-if) # standby 1 ip
Switch(config-if) # standby 1 timers 5 15
Switch(config-if) # end
```

Configuring HSRP Groups and Clustering: Example

This example shows how to bind standby group my_hsrp to the cluster and enable the same HSRP group to be used for command switch redundancy and router redundancy. The command can only be executed on the

cluster command switch. If the standby group name or number does not exist, or if the switch is a cluster member switch, an error message appears.

```
Switch # configure terminal
Switch(config) # cluster standby-group my_hsrp routing-redundancy
Switch(config-if) # end
```

Information About VRRP

Configuring VRRP

Virtual Router Redundancy Protocol (VRRP) is an election protocol that enables a group of routers to form a single virtual router to provide redundancy. In a VRRP configuration, one router is elected as the virtual router primary, and the other routers act as backups in case it fails. The LAN clients can then be configured with the virtual router as their default gateway, allowing several routers on a multi-access link to use the same virtual IP address. The virtual router, representing a group of routers, forms a VRRP group.

Both HSRP and VRRP perform the same function. You can choose to configure either IETF standard VRRP or Cisco's more powerful HSRP protocol on a device or stack.

Restrictions for VRRP

- The VRRP implementation on the switch does not support the MIB specified in RFC 2787.
- The VRRP implementation on the switch supports only text-based authentication.
- VRRPv3 is not supported.

Additional References for Configuring HSRP

Related Documents

Related Topic	Document Title
Cisco IOS commands	Cisco IOS Master Commands List, All Releases

Standards and RFCs

Standard/RFC	Title
<i>RFC 2281</i>	Cisco Hot Standby Router Protocol

MIBs

MIB	MIBs Link
All the supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
<p>The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies.</p> <p>To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds.</p> <p>Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.</p>	http://www.cisco.com/support

Feature Information for HSRP

Release	Modification
Cisco IOS 15.0(2)EX1	This feature was introduced.



PART **IV**

IPv6

- [Configuring MLD Snooping, on page 307](#)
- [Configuring IPv6 Unicast Routing, on page 323](#)
- [Implementing IPv6 Multicast, on page 337](#)
- [Configuring IPv6 ACL, on page 365](#)



CHAPTER 21

Configuring MLD Snooping

This module contains details of configuring MLD snooping

- [Information About Configuring IPv6 MLD Snooping, on page 307](#)
- [How to Configure IPv6 MLD Snooping, on page 311](#)
- [Displaying MLD Snooping Information, on page 319](#)
- [Configuration Examples for Configuring MLD Snooping, on page 320](#)

Information About Configuring IPv6 MLD Snooping



Note To use IPv6 MLD Snooping, the switch must be running the LAN Base image.

You can use Multicast Listener Discovery (MLD) snooping to enable efficient distribution of IP Version 6 (IPv6) multicast data to clients and routers in a switched network on the switch. Unless otherwise noted, the term switch refers to a standalone switch and to a switch stack.



Note Stacking is supported only on Catalyst 2960-X switches running the LAN base image.



Note To use IPv6, you must configure the dual IPv4 and IPv6 Switch Database Management (SDM) template on the switch.

On switches running the LAN Base feature set, the routing template is not supported.



Note For complete syntax and usage information for the commands used in this chapter, see the command reference for this release or the Cisco IOS documentation referenced in the procedures.

Understanding MLD Snooping

In IP Version 4 (IPv4), Layer 2 switches can use Internet Group Management Protocol (IGMP) snooping to limit the flooding of multicast traffic by dynamically configuring Layer 2 interfaces so that multicast traffic is forwarded to only those interfaces associated with IP multicast devices. In IPv6, MLD snooping performs a similar function. With MLD snooping, IPv6 multicast data is selectively forwarded to a list of ports that want to receive the data, instead of being flooded to all ports in a VLAN. This list is constructed by snooping IPv6 multicast control packets.

MLD is a protocol used by IPv6 multicast routers to discover the presence of multicast listeners (nodes wishing to receive IPv6 multicast packets) on the links that are directly attached to the routers and to discover which multicast packets are of interest to neighboring nodes. MLD is derived from IGMP; MLD Version 1 (MLDv1) is equivalent to IGMPv2, and MLD Version 2 (MLDv2) is equivalent to IGMPv3. MLD is a subprotocol of Internet Control Message Protocol Version 6 (ICMPv6), and MLD messages are a subset of ICMPv6 messages, identified in IPv6 packets by a preceding Next Header value of 58.

The switch supports two versions of MLD snooping:

- MLDv1 snooping detects MLDv1 control packets and sets up traffic bridging based on IPv6 destination multicast addresses.
- MLDv2 basic snooping (MBSS) uses MLDv2 control packets to set up traffic forwarding based on IPv6 destination multicast addresses.

The switch can snoop on both MLDv1 and MLDv2 protocol packets and bridge IPv6 multicast data based on destination IPv6 multicast addresses.



Note The switch does not support MLDv2 enhanced snooping, which sets up IPv6 source and destination multicast address-based forwarding.

MLD snooping can be enabled or disabled globally or per VLAN. When MLD snooping is enabled, a per-VLAN IPv6 multicast address table is constructed in software and hardware. The switch then performs IPv6 multicast-address based bridging in hardware.

MLD Messages

MLDv1 supports three types of messages:

- Listener Queries are the equivalent of IGMPv2 queries and are either General Queries or Multicast-Address-Specific Queries (MASQs).
- Multicast Listener Reports are the equivalent of IGMPv2 reports
- Multicast Listener Done messages are the equivalent of IGMPv2 leave messages.

MLDv2 supports MLDv2 queries and reports, as well as MLDv1 Report and Done messages.

Message timers and state transitions resulting from messages being sent or received are the same as those of IGMPv2 messages. MLD messages that do not have valid link-local IPv6 source addresses are ignored by MLD routers and switches.

MLD Queries

The switch sends out MLD queries, constructs an IPv6 multicast address database, and generates MLD group-specific and MLD group-and-source-specific queries in response to MLD Done messages. The switch also supports report suppression, report proxying, Immediate-Leave functionality, and static IPv6 multicast group address configuration.

When MLD snooping is disabled, all MLD queries are flooded in the ingress VLAN.

When MLD snooping is enabled, received MLD queries are flooded in the ingress VLAN, and a copy of the query is sent to the CPU for processing. From the received query, MLD snooping builds the IPv6 multicast address database. It detects multicast router ports, maintains timers, sets report response time, learns the querier IP source address for the VLAN, learns the querier port in the VLAN, and maintains multicast-address aging.



Note When the IPv6 multicast router is a Catalyst 6500 switch and you are using extended VLANs (in the range 1006 to 4094), IPv6 MLD snooping must be enabled on the extended VLAN on the Catalyst 6500 switch in order for the Catalyst 2960, 2960-S, 2960-C, 2960-X or 2960-CX switch to receive queries on the VLAN. For normal-range VLANs (1 to 1005), it is not necessary to enable IPv6 MLD snooping on the VLAN on the Catalyst 6500 switch.

When a group exists in the MLD snooping database, the switch responds to a group-specific query by sending an MLDv1 report. When the group is unknown, the group-specific query is flooded to the ingress VLAN.

When a host wants to leave a multicast group, it can send out an MLD Done message (equivalent to IGMP Leave message). When the switch receives an MLDv1 Done message, if Immediate-Leave is not enabled, the switch sends an MASQ to the port from which the message was received to determine if other devices connected to the port should remain in the multicast group.

Multicast Client Aging Robustness

You can configure port membership removal from addresses based on the number of queries. A port is removed from membership to an address only when there are no reports to the address on the port for the configured number of queries. The default number is 2.

Multicast Router Discovery

Like IGMP snooping, MLD snooping performs multicast router discovery, with these characteristics:

- Ports configured by a user never age out.
- Dynamic port learning results from MLDv1 snooping queries and IPv6 PIMv2 packets.
- If there are multiple routers on the same Layer 2 interface, MLD snooping tracks a single multicast router on the port (the router that most recently sent a router control packet).
- Dynamic multicast router port aging is based on a default timer of 5 minutes; the multicast router is deleted from the router port list if no control packet is received on the port for 5 minutes.
- IPv6 multicast router discovery only takes place when MLD snooping is enabled on the switch.
- Received IPv6 multicast router control packets are always flooded to the ingress VLAN, whether or not MLD snooping is enabled on the switch.

- After the discovery of the first IPv6 multicast router port, unknown IPv6 multicast data is forwarded only to the discovered router ports (before that time, all IPv6 multicast data is flooded to the ingress VLAN).

MLD Reports

The processing of MLDv1 join messages is essentially the same as with IGMPv2. When no IPv6 multicast routers are detected in a VLAN, reports are not processed or forwarded from the switch. When IPv6 multicast routers are detected and an MLDv1 report is received, an IPv6 multicast group address is entered in the VLAN MLD database. Then all IPv6 multicast traffic to the group within the VLAN is forwarded using this address. When MLD snooping is disabled, reports are flooded in the ingress VLAN.

When MLD snooping is enabled, MLD report suppression, called listener message suppression, is automatically enabled. With report suppression, the switch forwards the first MLDv1 report received by a group to IPv6 multicast routers; subsequent reports for the group are not sent to the routers. When MLD snooping is disabled, report suppression is disabled, and all MLDv1 reports are flooded to the ingress VLAN.

The switch also supports MLDv1 proxy reporting. When an MLDv1 MASQ is received, the switch responds with MLDv1 reports for the address on which the query arrived if the group exists in the switch on another port and if the port on which the query arrived is not the last member port for the address.

MLD Done Messages and Immediate-Leave

When the Immediate-Leave feature is enabled and a host sends an MLDv1 Done message (equivalent to an IGMP leave message), the port on which the Done message was received is immediately deleted from the group. You enable Immediate-Leave on VLANs and (as with IGMP snooping), you should only use the feature on VLANs where a single host is connected to the port. If the port was the last member of a group, the group is also deleted, and the leave information is forwarded to the detected IPv6 multicast routers.

When Immediate Leave is not enabled in a VLAN (which would be the case when there are multiple clients for a group on the same port) and a Done message is received on a port, an MASQ is generated on that port. The user can control when a port membership is removed for an existing address in terms of the number of MASQs. A port is removed from membership to an address when there are no MLDv1 reports to the address on the port for the configured number of queries.

The number of MASQs generated is configured by using the **ipv6 mld snooping last-listener-query count** global configuration command. The default number is 2.

The MASQ is sent to the IPv6 multicast address for which the Done message was sent. If there are no reports sent to the IPv6 multicast address specified in the MASQ during the switch maximum response time, the port on which the MASQ was sent is deleted from the IPv6 multicast address database. The maximum response time is the time configured by using the **ipv6 mld snooping last-listener-query-interval** global configuration command. If the deleted port is the last member of the multicast address, the multicast address is also deleted, and the switch sends the address leave information to all detected multicast routers.

Topology Change Notification Processing

When topology change notification (TCN) solicitation is enabled by using the **ipv6 mld snooping tcn query solicit** global configuration command, MLDv1 snooping sets the VLAN to flood all IPv6 multicast traffic with a configured number of MLDv1 queries before it begins sending multicast data only to selected ports. You set this value by using the **ipv6 mld snooping tcn flood query count** global configuration command. The default is to send two queries. The switch also generates MLDv1 global Done messages with valid link-local IPv6 source addresses when the switch becomes the STP root in the VLAN or when it is configured by the user. This is same as done in IGMP snooping.

MLD Snooping in Switch Stacks

The MLD IPv6 group address databases are maintained on all switches in the stack, regardless of which switch learns of an IPv6 multicast group. Report suppression and proxy reporting are done stack-wide. During the maximum response time, only one received report for a group is forwarded to the multicast routers, regardless of which switch the report arrives on.

The election of a new active stack does not affect the learning or bridging of IPv6 multicast data; bridging of IPv6 multicast data does not stop during an active stack re-election. When a new switch is added to the stack, it synchronizes the learned IPv6 multicast information from the active stack. Until the synchronization is complete, data ingress on the newly added switch is treated as unknown multicast data.

How to Configure IPv6 MLD Snooping

Default MLD Snooping Configuration

Table 37: Default MLD Snooping Configuration

Feature	Default Setting
MLD snooping (Global)	Disabled.
MLD snooping (per VLAN)	Enabled. MLD snooping must be globally enabled for VLAN MLD snooping to take place.
IPv6 Multicast addresses	None configured.
IPv6 Multicast router ports	None configured.
MLD snooping Immediate Leave	Disabled.
MLD snooping robustness variable	Global: 2; Per VLAN: 0. Note The VLAN value overrides the global setting. When the VLAN value is 0, the VLAN uses the global count.
Last listener query count	Global: 2; Per VLAN: 0. Note The VLAN value overrides the global setting. When the VLAN value is 0, the VLAN uses the global count.
Last listener query interval	Global: 1000 (1 second); VLAN: 0. Note The VLAN value overrides the global setting. When the VLAN value is 0, the VLAN uses the global interval.
TCN query solicit	Disabled.
TCN query count	2.

Feature	Default Setting
MLD listener suppression	

MLD Snooping Configuration Guidelines

When configuring MLD snooping, consider these guidelines:

- You can configure MLD snooping characteristics at any time, but you must globally enable MLD snooping by using the **ipv6 mld snooping** global configuration command for the configuration to take effect.
- When the IPv6 multicast router is a Catalyst 6500 switch and you are using extended VLANs (in the range 1006 to 4094), IPv6 MLD snooping must be enabled on the extended VLAN on the Catalyst 6500 switch in order for the switch to receive queries on the VLAN. For normal-range VLANs (1 to 1005), it is not necessary to enable IPv6 MLD snooping on the VLAN on the Catalyst 6500 switch.
- MLD snooping and IGMP snooping act independently of each other. You can enable both features at the same time on the switch.
- The maximum number of multicast entries allowed on the switch or switch stack is determined by the configured SDM template.
- The maximum number of address entries allowed for the switch or switch stack is 1000.

Enabling or Disabling MLD Snooping on the Switch

By default, IPv6 MLD snooping is globally disabled on the switch and enabled on all VLANs. When MLD snooping is globally disabled, it is also disabled on all VLANs. When you globally enable MLD snooping, the VLAN configuration overrides the global configuration. That is, MLD snooping is enabled only on VLAN interfaces in the default state (enabled).

You can enable and disable MLD snooping on a per-VLAN basis or for a range of VLANs, but if you globally disable MLD snooping, it is disabled in all VLANs. If global snooping is enabled, you can enable or disable VLAN snooping.

To globally enable MLD snooping on the switch, perform this procedure:

Procedure

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.

	Command or Action	Purpose
Step 3	ipv6 mld snooping Example: Device(config)# ipv6 mld snooping	Enables MLD snooping on the switch.
Step 4	end Example: Device(config)# end	Returns to privileged EXEC mode.
Step 5	copy running-config startup-config Example: Device(config)# copy running-config startup-config	(Optional) Save your entries in the configuration file.
Step 6	reload Example: Device(config)# reload	Reload the operating system.

Enabling or Disabling MLD Snooping on a VLAN

To enable MLD snooping on a VLAN, perform this procedure:

Procedure

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	ipv6 mld snooping Example:	Enables MLD snooping on the switch.

	Command or Action	Purpose
	Device(config) # <code>ipv6 mld snooping</code>	
Step 4	ipv6 mld snooping vlan <i>vlan-id</i> Example: Device(config) # <code>ipv6 mld snooping vlan 1</code>	Enables MLD snooping on the VLAN. The VLAN ID range is 1 to 1001 and 1006 to 4094. Note MLD snooping must be globally enabled for VLAN snooping to be enabled.
Step 5	end Example: Device(config) # <code>ipv6 mld snooping vlan 1</code>	Returns to privileged EXEC mode.

Configuring a Static Multicast Group

Hosts or Layer 2 ports normally join multicast groups dynamically, but you can also statically configure an IPv6 multicast address and member ports for a VLAN.

To add a Layer 2 port as a member of a multicast group, perform this procedure:

Procedure

	Command or Action	Purpose
Step 1	enable Example: Device> <code>enable</code>	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode.
Step 3	ipv6 mld snooping vlan <i>vlan-id</i> static <i>ipv6_multicast_address</i> interface <i>interface-id</i> Example: Device(config) # <code>ipv6 mld snooping vlan 1 static 3333.0000.1111 interface gigabitethernet 0/1</code>	Configures a multicast group with a Layer 2 port as a member of a multicast group: <ul style="list-style-type: none"> • <i>vlan-id</i> is the multicast group VLAN ID. The VLAN ID range is 1 to 1001 and 1006 to 4094. • <i>ipv6_multicast_address</i> is the 128-bit group IPv6 address. The address must be in the form specified in RFC 2373. • <i>interface-id</i> is the member port. It can be a physical interface or a port channel (1 to 48).

	Command or Action	Purpose
Step 4	end Example: Device(config)# end	Returns to privileged EXEC mode.
Step 5	Use one of the following: <ul style="list-style-type: none"> • show ipv6 mld snooping address • show ipv6 mld snooping address vlan <i>vlan-id</i> Example: Device# show ipv6 mld snooping address or Device# show ipv6 mld snooping vlan 1	Verifies the static member port and the IPv6 address.

Configuring a Multicast Router Port



Note Static connections to multicast routers are supported only on switch ports.

To add a multicast router port to a VLAN, perform this procedure:

Procedure

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	ipv6 mld snooping vlan <i>vlan-id</i> mrouter interface <i>interface-id</i> Example: Device(config)# ipv6 mld snooping vlan 1 mrouter interface gigabitethernet 0/2	Specifies the multicast router VLAN ID, and specify the interface to the multicast router. <ul style="list-style-type: none"> • The VLAN ID range is 1 to 1001 and 1006 to 4094. • The interface can be a physical interface or a port channel. The port-channel range is 1 to 48.

	Command or Action	Purpose
Step 4	end Example: Device(config)# end	Returns to privileged EXEC mode.
Step 5	show ipv6 mld snooping mrouter [vlan <i>vlan-id</i>] Example: Device# show ipv6 mld snooping mrouter vlan 1	Verifies that IPv6 MLD snooping is enabled on the VLAN interface.

Enabling MLD Immediate Leave

To enable MLDv1 immediate leave, perform this procedure:

Procedure

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	ipv6 mld snooping vlan <i>vlan-id</i> immediate-leave Example: Device(config)# ipv6 mld snooping vlan 1 immediate-leave	Enables MLD Immediate Leave on the VLAN interface.
Step 4	end Example: Device(config)# end	Returns to privileged EXEC mode.
Step 5	show ipv6 mld snooping vlan <i>vlan-id</i> Example: Device# show ipv6 mld snooping vlan 1	Verifies that Immediate Leave is enabled on the VLAN interface.

Configuring MLD Snooping Queries

To configure MLD snooping query characteristics for the switch or for a VLAN, perform this procedure:

Procedure

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	ipv6 mld snooping robustness-variable <i>value</i> Example: Device(config)# ipv6 mld snooping robustness-variable 3	(Optional) Sets the number of queries that are sent before switch will delete a listener (port) that does not respond to a general query. The range is 1 to 3; the default is 2.
Step 4	ipv6 mld snooping vlan <i>vlan-id</i> robustness-variable <i>value</i> Example: Device(config)# ipv6 mld snooping vlan 1 robustness-variable 3	(Optional) Sets the robustness variable on a VLAN basis, which determines the number of general queries that MLD snooping sends before aging out a multicast address when there is no MLD report response. The range is 1 to 3; the default is 0. When set to 0, the number used is the global robustness variable value.
Step 5	ipv6 mld snooping last-listener-query-count <i>count</i> Example: Device(config)# ipv6 mld snooping last-listener-query-count 7	(Optional) Sets the number of MASQs that the switch sends before aging out an MLD client. The range is 1 to 7; the default is 2. The queries are sent 1 second apart.
Step 6	ipv6 mld snooping vlan <i>vlan-id</i> last-listener-query-count <i>count</i> Example: Device(config)# ipv6 mld snooping vlan 1 last-listener-query-count 7	(Optional) Sets the last-listener query count on a VLAN basis. This value overrides the value configured globally. The range is 1 to 7; the default is 0. When set to 0, the global count value is used. Queries are sent 1 second apart.
Step 7	ipv6 mld snooping last-listener-query-interval <i>interval</i> Example: Device(config)# ipv6 mld snooping last-listener-query-interval 2000	(Optional) Sets the maximum response time that the switch waits after sending out a MASQ before deleting a port from the multicast group. The range is 100 to 32,768 thousands of a second. The default is 1000 (1 second).

	Command or Action	Purpose
Step 8	ipv6 mld snooping vlan <i>vlan-id</i> last-listener-query-interval <i>interval</i> Example: Device(config)# ipv6 mld snooping vlan 1 last-listener-query-interval 2000	(Optional) Sets the last-listener query interval on a VLAN basis. This value overrides the value configured globally. The range is 0 to 32,768 thousands of a second. The default is 0. When set to 0, the global last-listener query interval is used.
Step 9	ipv6 mld snooping tcn query solicit Example: Device(config)# ipv6 mld snooping tcn query solicit	(Optional) Enables topology change notification (TCN) solicitation, which means that VLANs flood all IPv6 multicast traffic for the configured number of queries before sending multicast data to only those ports requesting to receive it. The default is for TCN to be disabled.
Step 10	ipv6 mld snooping tcn flood query count <i>count</i> Example: Device(config)# ipv6 mld snooping tcn flood query count 5	(Optional) When TCN is enabled, specifies the number of TCN queries to be sent. The range is from 1 to 10; the default is 2.
Step 11	end	Returns to privileged EXEC mode.
Step 12	show ipv6 mld snooping querier [vlan <i>vlan-id</i>] Example: Device(config)# show ipv6 mld snooping querier vlan 1	(Optional) Verifies that the MLD snooping querier information for the switch or for the VLAN.

Disabling MLD Listener Message Suppression

MLD snooping listener message suppression is enabled by default. When it is enabled, the switch forwards only one MLD report per multicast router query. When message suppression is disabled, multiple MLD reports could be forwarded to the multicast routers.

To disable MLD listener message suppression, perform this procedure:

Procedure

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enter global configuration mode.

	Command or Action	Purpose
Step 3	no ipv6 mld snooping listener-message-suppression Example: Device(config)# no ipv6 mld snooping listener-message-suppression	Disable MLD message suppression.
Step 4	end Example: Device(config)# end	Return to privileged EXEC mode.
Step 5	show ipv6 mld snooping Example: Device# show ipv6 mld snooping	Verify that IPv6 MLD snooping report suppression is disabled.

Displaying MLD Snooping Information

You can display MLD snooping information for dynamically learned and statically configured router ports and VLAN interfaces. You can also display IPv6 group address multicast entries for a VLAN configured for MLD snooping.

Table 38: Commands for Displaying MLD Snooping Information

Command	Purpose
show ipv6 mld snooping [vlan <i>vlan-id</i>]	Displays the MLD snooping configuration information for all VLANs on the switch or for a specified VLAN. (Optional) Enter vlan <i>vlan-id</i> to display information for a single VLAN. The VLAN ID range is 1 to 1001 and 1006 to 4094.
show ipv6 mld snooping mrouter [vlan <i>vlan-id</i>]	Displays information on dynamically learned and manually configured multicast router interfaces. When you enable MLD snooping, the switch automatically learns the interface to which a multicast router is connected. These are dynamically learned interfaces. (Optional) Enters vlan <i>vlan-id</i> to display information for a single VLAN. The VLAN ID range is 1 to 1001 and 1006 to 4094.
show ipv6 mld snooping querier [vlan <i>vlan-id</i>]	Displays information about the IPv6 address and incoming port for the most-recently received MLD query messages in the VLAN. (Optional) Enters vlan <i>vlan-id</i> to display information for a single VLAN. The VLAN ID range is 1 to 1001 and 1006 to 4094.

Command	Purpose
show ipv6 mld snooping address [vlan <i>vlan-id</i>] [count dynamic user]	Displays all IPv6 multicast address information or specific IPv6 multicast address information for the switch or a VLAN. <ul style="list-style-type: none"> • Enters count to show the group count on the switch or in a VLAN. • Enters dynamic to display MLD snooping learned group information for the switch or for a VLAN. • Enters user to display MLD snooping user-configured group information for the switch or for a VLAN.
show ipv6 mld snooping address vlan <i>vlan-id</i> [<i>ipv6-multicast-address</i>]	Displays MLD snooping for the specified VLAN and IPv6 multicast address.

Configuration Examples for Configuring MLD Snooping

Configuring a Static Multicast Group: Example

This example shows how to statically configure an IPv6 multicast group:

```
Device# configure terminal
Device(config)# ipv6 mld snooping vlan 2 static 3333.0000.1111 interface gigabitethernet1/0/1
Device(config)# end
```

Configuring a Multicast Router Port: Example

This example shows how to add a multicast router port to VLAN 200:

```
Device# configure terminal
Device(config)# ipv6 mld snooping vlan 200 mrouter interface gigabitethernet
0/2
Device(config)# exit
```

Enabling MLD Immediate Leave: Example

This example shows how to enable MLD Immediate Leave on VLAN 130:

```
Device# configure terminal
Device(config)# ipv6 mld snooping vlan 130 immediate-leave
Device(config)# exit
```

Configuring MLD Snooping Queries: Example

This example shows how to set the MLD snooping global robustness variable to 3:

```
Device# configure terminal  
Device(config)# ipv6 mld snooping robustness-variable 3  
Device(config)# exit
```

This example shows how to set the MLD snooping last-listener query count for a VLAN to 3:

```
Device# configure terminal  
Device(config)# ipv6 mld snooping vlan 200 last-listener-query-count 3  
Device(config)# exit
```

This example shows how to set the MLD snooping last-listener query interval (maximum response time) to 2000 (2 seconds):

```
Device# configure terminal  
Device(config)# ipv6 mld snooping last-listener-query-interval 2000  
Device(config)# exit
```




CHAPTER 22

Configuring IPv6 Unicast Routing

- [Information About Configuring IPv6 Host Functions](#) , on page 323
- [Configuration Examples for IPv6 Unicast Routing](#), on page 335

Information About Configuring IPv6 Host Functions

This chapter describes how to configure IPv6 host functions on the Catalyst 2960, 2960-S, and 2960-C.



Note To use IPv6 Host Functions, the switch must be running the LAN Base image.

For information about configuring IPv6 Multicast Listener Discovery (MLD) snooping, see *Configuring MLD Snooping*.

To enable dual stack environments (supporting both IPv4 and IPv6) on a Catalyst 2960 switch, you must configure the switch to use the a dual IPv4 and IPv6 switch database management (SDM) template. See the ["Dual IPv4 and IPv6 Protocol Stacks"](#) section. This template is not required on Catalyst 2960-S switches.



Note For complete syntax and usage information for the commands used in this chapter, see the Cisco IOS documentation referenced in the procedures.

Understanding IPv6

IPv4 users can move to IPv6 and receive services such as end-to-end security, quality of service (QoS), and globally unique addresses. The IPv6 address space reduces the need for private addresses and Network Address Translation (NAT) processing by border routers at network edges.

For information about how Cisco Systems implements IPv6, go to:

http://www.cisco.com/en/US/products/ps6553/products_ios_technology_home.html

For information about IPv6 and other features in this chapter

- See the *Cisco IOS IPv6 Configuration Library*.

- Use the Search field on Cisco.com to locate the Cisco IOS software documentation. For example, if you want information about static routes, you can enter *Implementing Static Routes for IPv6* in the search field to learn about static routes.

IPv6 Addresses

The switch supports only IPv6 unicast addresses. It does not support site-local unicast addresses, or anycast addresses.

The IPv6 128-bit addresses are represented as a series of eight 16-bit hexadecimal fields separated by colons in the format: n:n:n:n:n:n:n:n. This is an example of an IPv6 address:

```
2031:0000:130F:0000:0000:09C0:080F:130B
```

For easier implementation, leading zeros in each field are optional. This is the same address without leading zeros:

```
2031:0:130F:0:0:9C0:80F:130B
```

You can also use two colons (::) to represent successive hexadecimal fields of zeros, but you can use this short version only once in each address:

```
2031:0:130F::09C0:080F:130B
```

For more information about IPv6 address formats, address types, and the IPv6 packet header, see the http://www.cisco.com/c/en/us/td/docs/ios-xml/ios/ipv6_basic/configuration/xr-3e/ipv6b-xr-3e-book.html of *Cisco IOS IPv6 Configuration Library* on Cisco.com.

- IPv6 Address Formats
- IPv6 Address Type: Multicast
- IPv6 Address Output Display
- Simplified IPv6 Packet Header

Supported IPv6 Unicast Routing Features

These sections describe the IPv6 protocol features supported by the switch:

128-Bit Wide Unicast Addresses

The switch supports aggregatable global unicast addresses and link-local unicast addresses. It does not support site-local unicast addresses.

- Aggregatable global unicast addresses are IPv6 addresses from the aggregatable global unicast prefix. The address structure enables strict aggregation of routing prefixes and limits the number of routing table entries in the global routing table. These addresses are used on links that are aggregated through organizations and eventually to the Internet service provider.

These addresses are defined by a global routing prefix, a subnet ID, and an interface ID. Current global unicast address allocation uses the range of addresses that start with binary value 001 (2000::/3). Addresses with a prefix of 2000::/3(001) through E000::/3(111) must have 64-bit interface identifiers in the extended unique identifier (EUI)-64 format.

- Link local unicast addresses can be automatically configured on any interface by using the link-local prefix FE80::/10(1111 1110 10) and the interface identifier in the modified EUI format. Link-local addresses are used in the neighbor discovery protocol (NDP) and the stateless autoconfiguration process.

Nodes on a local link use link-local addresses and do not require globally unique addresses to communicate. IPv6 routers do not forward packets with link-local source or destination addresses to other links.

For more information, see the section about IPv6 unicast addresses in the “Implementing IPv6 Addressing and Basic Connectivity” chapter in the *Cisco IOS IPv6 Configuration Library* on Cisco.com.

DNS for IPv6

IPv6 supports Domain Name System (DNS) record types in the DNS name-to-address and address-to-name lookup processes. The DNS AAAA resource record types support IPv6 addresses and are equivalent to an A address record in IPv4. The switch supports DNS resolution for IPv4 and IPv6.

ICMPv6

The Internet Control Message Protocol (ICMP) in IPv6 generates error messages, such as ICMP destination unreachable messages, to report errors during processing and other diagnostic functions. In IPv6, ICMP packets are also used in the neighbor discovery protocol and path MTU discovery.

Neighbor Discovery

The switch supports NDP for IPv6, a protocol running on top of ICMPv6, and static neighbor entries for IPv6 stations that do not support NDP. The IPv6 neighbor discovery process uses ICMP messages and solicited-node multicast addresses to determine the link-layer address of a neighbor on the same network (local link), to verify the reachability of the neighbor, and to keep track of neighboring routers.

The switch supports ICMPv6 redirect for routes with mask lengths less than 64 bits. ICMP redirect is not supported for host routes or for summarized routes with mask lengths greater than 64 bits.

Neighbor discovery throttling ensures that the switch CPU is not unnecessarily burdened while it is in the process of obtaining the next hop forwarding information to route an IPv6 packet. The switch drops any additional IPv6 packets whose next hop is the same neighbor that the switch is actively trying to resolve. This drop avoids further load on the CPU.

IPv6 Stateless Autoconfiguration and Duplicate Address Detection

The switch uses stateless autoconfiguration to manage link, subnet, and site addressing changes, such as management of host and mobile IP addresses. A host autonomously configures its own link-local address, and booting nodes send router solicitations to request router advertisements for configuring interfaces.

For more information about autoconfiguration and duplicate address detection, see the “Implementing IPv6 Addressing and Basic Connectivity” chapter of *Cisco IOS IPv6 Configuration Library* on Cisco.com.

IPv6 Applications

The switch has IPv6 support for these applications:

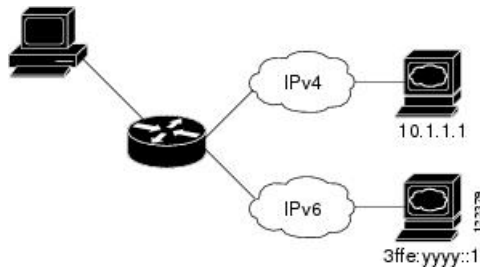
- Ping, traceroute, Telnet
- Secure Shell (SSH) over an IPv6 transport
- HTTP server access over IPv6 transport
- DNS resolver for AAAA over IPv4 transport
- Cisco Discovery Protocol (CDP) support for IPv6 addresses

For more information about managing these applications, see the *Cisco IOS IPv6 Configuration Library* on Cisco.com.

Dual IPv4 and IPv6 Protocol Stacks

This figure shows a router forwarding both IPv4 and IPv6 traffic through the same interface, based on the IP packet and destination addresses.

Figure 18: Dual IPv4 and IPv6 Support on an Interface



Use the dual IPv4 and IPv6 switch database management (SDM) template to enable IPv6 routing dual stack environments (supporting both IPv4 and IPv6). For more information about the dual IPv4 and IPv6 SDM template, see *Configuring SDM Templates*.

The dual IPv4 and IPv6 templates allow the switch to be used in dual stack environments.

- If you try to configure IPv6 without first selecting a dual IPv4 and IPv6 template, a warning message appears.
- In IPv4-only environments, the switch routes IPv4 packets and applies IPv4 QoS and ACLs in hardware. IPv6 packets are not supported.
- In dual IPv4 and IPv6 environments, the switch applies IPv4 QoS and ACLs in hardware .
- If you do not plan to use IPv6, do not use the dual stack template because this template results in less hardware memory capacity for each resource.

For more information about IPv4 and IPv6 protocol stacks, see the “Implementing IPv6 Addressing and Basic Connectivity” chapter of *Cisco IOS IPv6 Configuration Library* on Cisco.com.

SNMP and Syslog Over IPv6

To support both IPv4 and IPv6, IPv6 network management requires both IPv6 and IPv4 transports. Syslog over IPv6 supports address data types for these transports.

Simple Network Management Protocol (SNMP) and syslog over IPv6 provide these features:

- Support for both IPv4 and IPv6
- IPv6 transport for SNMP and to modify the SNMP agent to support traps for an IPv6 host
- SNMP- and syslog-related MIBs to support IPv6 addressing
- Configuration of IPv6 hosts as trap receivers

For support over IPv6, SNMP modifies the existing IP transport mapping to simultaneously support IPv4 and IPv6. These SNMP actions support IPv6 transport management:

- Opens User Datagram Protocol (UDP) SNMP socket with default settings

- Provides a new transport mechanism called *SR_IPV6_TRANSPORT*
- Sends SNMP notifications over IPv6 transport
- Supports SNMP-named access lists for IPv6 transport
- Supports SNMP proxy forwarding using IPv6 transport
- Verifies SNMP Manager feature works with IPv6 transport

For information on SNMP over IPv6, including configuration procedures, see the “Managing Cisco IOS Applications over IPv6” chapter in the *Cisco IOS IPv6 Configuration Library* on Cisco.com.

For information about syslog over IPv6, including configuration procedures, see the “Implementing IPv6 Addressing and Basic Connectivity” chapter in the *Cisco IOS IPv6 Configuration Library* on Cisco.com.

HTTP(S) Over IPv6

The HTTP client sends requests to both IPv4 and IPv6 HTTP servers, which respond to requests from both IPv4 and IPv6 HTTP clients. URLs with literal IPv6 addresses must be specified in hexadecimal using 16-bit values between colons.

The accept socket call chooses an IPv4 or IPv6 address family. The accept socket is either an IPv4 or IPv6 socket. The listening socket continues to listen for both IPv4 and IPv6 signals that indicate a connection. The IPv6 listening socket is bound to an IPv6 wildcard address.

The underlying TCP/IP stack supports a dual-stack environment. HTTP relies on the TCP/IP stack and the sockets for processing network-layer interactions.

Basic network connectivity (**ping**) must exist between the client and the server hosts before HTTP connections can be made.

For more information, see the “Managing Cisco IOS Applications over IPv6” chapter in the *Cisco IOS IPv6 Configuration Library* on Cisco.com.

EIGRP IPv6

Switches support the Enhanced Interior Gateway Routing Protocol (EIGRP) for IPv6. It is configured on the interfaces on which it runs and does not require a global IPv6 address. Switches running IP Lite only support EIGRPv6 stub routing.

Before running, an instance of EIGRP IPv6 requires an implicit or explicit router ID. An implicit router ID is derived from a local IPv6 address, so any IPv6 node always has an available router ID. However, EIGRP IPv6 might be running in a network with only IPv6 nodes and therefore might not have an available IPv6 router ID.

For more information about EIGRP for IPv6, see the “Implementing EIGRP for IPv6” chapter in the *Cisco IOS IPv6 Configuration Library* on Cisco.com.

EIGRPv6 Stub Routing

The EIGRPv6 stub routing feature, reduces resource utilization by moving routed traffic closer to the end user.

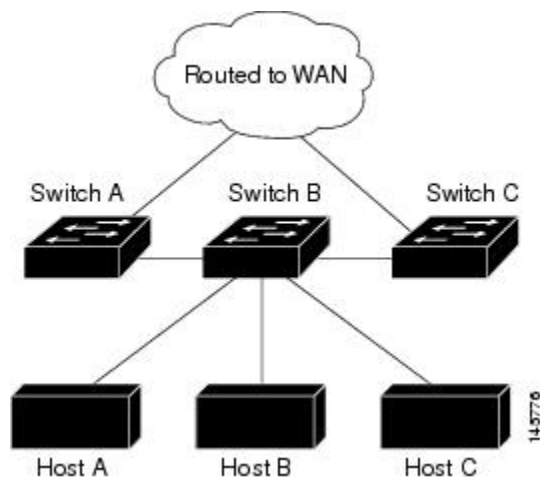
In a network using EIGRPv6 stub routing, the only allowable route for IPv6 traffic to the user is through a switch that is configured with EIGRPv6 stub routing. The switch sends the routed traffic to interfaces that are configured as user interfaces or are connected to other devices.

When using EIGRPv6 stub routing, you need to configure the distribution and remote routers to use EIGRPv6 and to configure only the switch as a stub. Only specified routes are propagated from the switch. The switch responds to all queries for summaries, connected routes, and routing updates.

Any neighbor that receives a packet informing it of the stub status does not query the stub router for any routes, and a router that has a stub peer does not query that peer. The stub router depends on the distribution router to send the proper updates to all peers.

In the figure given below, switch B is configured as an EIGRPv6 stub router. Switches A and C are connected to the rest of the WAN. Switch B advertises connected, static, redistribution, and summary routes to switch A and C. Switch B does not advertise any routes learned from switch A (and the reverse).

Figure 19: EIGRP Stub Router Configuration



For more information about EIGRPv6 stub routing, see “Implementing EIGRP for IPv6” section of the *Cisco IOS IP Configuration Guide, Volume 2 of 3: Routing Protocols, Release 12.4*.

IPv6 and Switch Stacks

The switch supports IPv6 forwarding across the stack and IPv6 host functionality on the active stack. The active stack runs IPv6 host functionality and IPv6 applications.

While the new active stack is being elected and is resetting, the switch stack does not forward IPv6 packets. The stack MAC address changes, which also changes the IPv6 address. When you specify the stack IPv6 address with an extended unique identifier (EUI) by using the **ipv6 address** ipv6-prefix/prefix length eui-64 interface configuration command, the address is based on the interface MAC address. See the ["Configuring IPv6 Addressing and Enabling IPv6 Host"](#) section.

If you configure the persistent MAC address feature on the stack and the active stack changes, the stack MAC address does not change for approximately 4 minutes. For more information, see the "Enabling Persistent MAC Address" section in "Managing Switch Stacks."

Default IPv6 Configuration

Table 39: Default IPv6 Configuration

Feature	Default Setting
SDM template	Advance desktop. Default is advanced template
IPv6 addresses	None configured

Configuring IPv6 Addressing and Enabling IPv6 Routing

This section describes how to assign IPv6 addresses to individual Layer 3 interfaces and to globally forward IPv6 traffic on the switch.

Before configuring IPv6 on the switch, consider these guidelines:

- Be sure to select a dual IPv4 and IPv6 SDM template.
- In the **ipv6 address** interface configuration command, you must enter the *ipv6-address* and *ipv6-prefix* variables with the address specified in hexadecimal using 16-bit values between colons. The *prefix-length* variable (preceded by a slash [/]) is a decimal value that shows how many of the high-order contiguous bits of the address comprise the prefix (the network portion of the address).

To forward IPv6 traffic on an interface, you must configure a global IPv6 address on that interface. Configuring an IPv6 address on an interface automatically configures a link-local address and activates IPv6 for the interface. The configured interface automatically joins these required multicast groups for that link:

- solicited-node multicast group FF02:0:0:0:0:1:ff00::/104 for each unicast address assigned to the interface (this address is used in the neighbor discovery process.)
- all-nodes link-local multicast group FF02::1
- all-routers link-local multicast group FF02::2

For more information about configuring IPv6 routing, see the “Implementing Addressing and Basic Connectivity for IPv6” chapter in the *Cisco IOS IPv6 Configuration Library* on Cisco.com.

Beginning in privileged EXEC mode, follow these steps to assign an IPv6 address to a Layer 3 interface and enable IPv6 forwarding:

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 2	sdm prefer dual-ipv4-and-ipv6 {default} Example:	Selects an SDM template that supports IPv4 and IPv6.

	Command or Action	Purpose
	Device (config) # sdm prefer dual-ipv4-and-ipv6 default	<ul style="list-style-type: none"> • default—Sets the switch to the default template to balance system resources.
Step 3	end Example: Device (config) # end	Returns to privileged EXEC mode.
Step 4	reload Example: Device# reload	Reloads the operating system.
Step 5	configure terminal Example: Device# configure terminal	Enters global configuration mode after the switch reloads.
Step 6	interface interface-id Example: Device (config) # interface gigabitethernet 1/0/1	Enters interface configuration mode, and specifies the Layer 3 interface to configure.
Step 7	Use one of the following: <ul style="list-style-type: none"> • ipv6 address ipv6-prefix/prefix length eui-64 • ipv6 address ipv6-address/prefix length • ipv6 address ipv6-address link-local • ipv6 enable Example: Device (config-if) # ipv6 address 2001:0DB8:c18:1::/64 eui 64 Device (config-if) # ipv6 address 2001:0DB8:c18:1::/64 Device (config-if) # ipv6 address 2001:0DB8:c18:1:: link-local Device (config-if) # ipv6 enable	<ul style="list-style-type: none"> • Specifies a global IPv6 address with an extended unique identifier (EUI) in the low-order 64 bits of the IPv6 address. Specify only the network prefix; the last 64 bits are automatically computed from the switch MAC address. This enables IPv6 processing on the interface. • Manually configures an IPv6 address on the interface. • Specifies a link-local address on the interface to be used instead of the link-local address that is automatically configured when IPv6 is enabled on the interface. This command enables IPv6 processing on the interface. • Automatically configures an IPv6 link-local address on the interface, and enables the interface for IPv6 processing. The link-local address can only be used to communicate with nodes on the same link.
Step 8	exit Example:	Returns to global configuration mode.

	Command or Action	Purpose
	Device(config-if)# exit	
Step 9	end Example: Device(config)# end	Returns to privileged EXEC mode.
Step 10	show ipv6 interface interface-id Example: Device# show ipv6 interface gigabitethernet 1/0/1	Verifies your entries.
Step 11	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Configuring IPv6 ICMP Rate Limiting

ICMP rate limiting is enabled by default with a default interval between error messages of 100 milliseconds and a bucket size (maximum number of tokens to be stored in a bucket) of 10.

To change the ICMP rate-limiting parameters, perform this procedure:

Procedure

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	ipv6 icmp error-interval interval [bucketsize] Example: Device(config)# ipv6 icmp error-interval 50 20	Configures the interval and bucket size for IPv6 ICMP error messages: <ul style="list-style-type: none"> <i>interval</i>—The interval (in milliseconds) between tokens being added to the bucket. The range is from 0 to 2147483647 milliseconds.

	Command or Action	Purpose
		<ul style="list-style-type: none"> <i>bucketsize</i>—(Optional) The maximum number of tokens stored in the bucket. The range is from 1 to 200.
Step 4	end Example: Device(config) # end	Returns to privileged EXEC mode.
Step 5	show ipv6 interface [<i>interface-id</i>] Example: Device# show ipv6 interface gigabitethernet0/1	Verifies your entries.
Step 6	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Configuring Static Routing for IPv6

For more information about configuring static IPv6 routing, see the “Implementing Static Routes for IPv6” chapter in the *Cisco IOS IPv6 Configuration Library* on Cisco.com.

To configure static IPv6 routing, perform this procedure:

Before you begin

You must enable routing by using the **ip routing** global configuration command, enable the forwarding of IPv6 packets by using the **ipv6 unicast-routing** global configuration command, and enable IPv6 on at least one Layer 3 interface by configuring an IPv6 address on the interface.

Procedure

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.

	Command or Action	Purpose
Step 3	<p>ipv6 route <i>ipv6-prefix/prefix length</i> {<i>ipv6-address</i> <i>interface-id</i> [<i>ipv6-address</i>]} [<i>administrative distance</i>]</p> <p>Example:</p> <pre>Device(config)# ipv6 route 2001:0DB8::/32 gigabitethernet2/0/1 130</pre>	<p>Configures a static IPv6 route.</p> <ul style="list-style-type: none"> • <i>ipv6-prefix</i>—The IPv6 network that is the destination of the static route. It can also be a hostname when static host routes are configured. • <i>/prefix length</i>—The length of the IPv6 prefix. A decimal value that shows how many of the high-order contiguous bits of the address comprise the prefix (the network portion of the address). A slash mark must precede the decimal value. • <i>ipv6-address</i>—The IPv6 address of the next hop that can be used to reach the specified network. The IPv6 address of the next hop need not be directly connected; recursion is done to find the IPv6 address of the directly connected next hop. The address must be in the form documented in RFC 2373, specified in hexadecimal using 16-bit values between colons. • <i>interface-id</i>—Specifies direct static routes from point-to-point and broadcast interfaces. With point-to-point interfaces, there is no need to specify the IPv6 address of the next hop. With broadcast interfaces, you should always specify the IPv6 address of the next hop, or ensure that the specified prefix is assigned to the link, specifying a link-local address as the next hop. You can optionally specify the IPv6 address of the next hop to which packets are sent. <p>Note You must specify an <i>interface-id</i> when using a link-local address as the next hop (the link-local next hop must also be an adjacent router).</p> <ul style="list-style-type: none"> • <i>administrative distance</i>—(Optional) An administrative distance. The range is 1 to 254; the default value is 1, which gives static routes precedence over any other type of route except connected routes. To configure a floating static route, use an administrative distance greater than that of the dynamic routing protocol.
Step 4	<p>end</p> <p>Example:</p> <pre>Device(config)# end</pre>	<p>Returns to privileged EXEC mode.</p>

	Command or Action	Purpose
Step 5	<p>Use one of the following:</p> <ul style="list-style-type: none"> • show ipv6 static [<i>ipv6-address</i> <i>ipv6-prefix/prefix length</i>] [interface <i>interface-id</i>] [detail][recursive] [detail] • show ipv6 route static [<i>updated</i>] <p>Example:</p> <pre>Device# show ipv6 static 2001:0DB8::/32 interface gigabitethernet2/0/1</pre> <p>OR</p> <pre>Device# show ipv6 route static</pre>	<p>Verifies your entries by displaying the contents of the IPv6 routing table.</p> <ul style="list-style-type: none"> • interface <i>interface-id</i>—(Optional) Displays only those static routes with the specified interface as an egress interface. • recursive—(Optional) Displays only recursive static routes. The recursive keyword is mutually exclusive with the interface keyword, but it can be used with or without the IPv6 prefix included in the command syntax. • detail—(Optional) Displays this additional information: <ul style="list-style-type: none"> • For valid recursive routes, the output path set, and maximum resolution depth. • For invalid routes, the reason why the route is not valid.
Step 6	<p>copy running-config startup-config</p> <p>Example:</p> <pre>Device# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Displaying IPv6

For complete syntax and usage information on these commands, see the Cisco IOS command reference publications.

Table 40: Command for Monitoring IPv6

Command	Purpose
show ipv6 access-list	Displays a summary of access lists.
show ipv6 cef	Displays Cisco Express Forwarding for IPv6.
show ipv6 interface <i>interface-id</i>	Displays IPv6 interface status and configuration.
show ipv6 mtu	Displays IPv6 MTU per destination cache.
show ipv6 neighbors	Displays IPv6 neighbor cache entries.
show ipv6 prefix-list	Displays a list of IPv6 prefix lists.
show ipv6 protocols	Displays a list of IPv6 routing protocols on the switch.

Command	Purpose
show ipv6 rip	Displays IPv6 RIP routing protocol status.
show ipv6 route	Displays IPv6 route table entries.
show ipv6 static	Displays IPv6 static routes.
show ipv6 traffic	Displays IPv6 traffic statistics.

Configuration Examples for IPv6 Unicast Routing

Configuring IPv6 Addressing and Enabling IPv6 Routing: Example

This example shows how to enable IPv6 with both a link-local address and a global address based on the IPv6 prefix 2001:0DB8:c18:1::/64. The EUI-64 interface ID is used in the low-order 64 bits of both addresses. Output from the **show ipv6 interface EXEC** command is included to show how the interface ID (20B:46FF:FE2F:D940) is appended to the link-local prefix FE80::/64 of the interface.

```
Device(config)# ipv6 unicast-routing
Device(config)# interface gigabitethernet0/11

Device(config-if)# ipv6 address 2001:0DB8:c18:1::/64 eui 64
Device(config-if)# end
Device# show ipv6 interface gigabitethernet0/11
GigabitEthernet0/11 is up, line protocol is up
  IPv6 is enabled, link-local address is FE80::20B:46FF:FE2F:D940
  Global unicast address(es):
    2001:0DB8:c18:1:20B:46FF:FE2F:D940, subnet is 2001:0DB8:c18:1::/64 [EUI]
  Joined group address(es):
    FE02::1
    FE02::2
    FE02::1:FF2F:D940
  MTU is 1500 bytes
  ICMP error messages limited to one every 100 milliseconds
  ICMP redirects are enabled
  ND DAD is enabled, number of DAD attempts: 1
  ND reachable time is 30000 milliseconds
  ND advertised reachable time is 0 milliseconds
  ND advertised retransmit interval is 0 milliseconds
  ND router advertisements are sent every 200 seconds
  ND router advertisements live for 1800 seconds
  Hosts use stateless autoconfig for addresses.
```

Configuring IPv6 ICMP Rate Limiting: Example

This example shows how to configure an IPv6 ICMP error message interval of 50 milliseconds and a bucket size of 20 tokens.

```
Device(config)#ipv6 icmp error-interval 50 20
```

Configuring Static Routing for IPv6: Example

This example shows how to configure a floating static route to an interface with an administrative distance of 130:

```
Device(config)# ipv6 route 2001:0DB8::/32 gigabitethernet 1/0/1 130
```

Displaying IPv6: Example

This is an example of the output from the **show ipv6 interface** privileged EXEC command:

```
Device# show ipv6 interface
Vlan1 is up, line protocol is up
IPv6 is enabled, link-local address is FE80::20B:46FF:FE2F:D940
Global unicast address(es):
  3FFE:C000:0:1:20B:46FF:FE2F:D940, subnet is 3FFE:C000:0:1::/64 [EUI]
Joined group address(es):
  FF02::1
  FF02::2
  FF02::1:FF2F:D940
MTU is 1500 bytes
ICMP error messages limited to one every 100 milliseconds
ICMP redirects are enabled
ND DAD is enabled, number of DAD attempts: 1
ND reachable time is 30000 milliseconds
ND advertised reachable time is 0 milliseconds
ND advertised retransmit interval is 0 milliseconds
ND router advertisements are sent every 200 seconds
ND router advertisements live for 1800 seconds
<output truncated>
```



CHAPTER 23

Implementing IPv6 Multicast

- [Information About Implementing IPv6 Multicast Routing, on page 337](#)
- [Implementing IPv6 Multicast, on page 343](#)

Information About Implementing IPv6 Multicast Routing

This chapter describes how to implement IPv6 multicast routing on the switch.

Traditional IP communication allows a host to send packets to a single host (unicast transmission) or to all hosts (broadcast transmission). IPv6 multicast provides a third scheme, allowing a host to send a single data stream to a subset of all hosts (group transmission) simultaneously.

IPv6 Multicast Overview

An IPv6 multicast group is an arbitrary group of receivers that want to receive a particular data stream. This group has no physical or geographical boundaries--receivers can be located anywhere on the Internet or in any private network. Receivers that are interested in receiving data flowing to a particular group must join the group by signaling their local switch. This signaling is achieved with the MLD protocol.

Switches use the MLD protocol to learn whether members of a group are present on their directly attached subnets. Hosts join multicast groups by sending MLD report messages. The network then delivers data to a potentially unlimited number of receivers, using only one copy of the multicast data on each subnet. IPv6 hosts that wish to receive the traffic are known as group members.

Packets delivered to group members are identified by a single multicast group address. Multicast packets are delivered to a group using best-effort reliability, just like IPv6 unicast packets.

The multicast environment consists of senders and receivers. Any host, regardless of whether it is a member of a group, can send to a group. However, only members of a group can listen to and receive the message.

A multicast address is chosen for the receivers in a multicast group. Senders use that address as the destination address of a datagram to reach all members of the group.

Membership in a multicast group is dynamic; hosts can join and leave at any time. There is no restriction on the location or number of members in a multicast group. A host can be a member of more than one multicast group at a time.

How active a multicast group is, its duration, and its membership can vary from group to group and from time to time. A group that has members may have no activity.

IPv6 Multicast Routing Implementation

The Cisco IOS software supports the following protocols to implement IPv6 multicast routing:

- MLD is used by IPv6 switches to discover multicast listeners (nodes that want to receive multicast packets destined for specific multicast addresses) on directly attached links. There are two versions of MLD: MLD version 1 is based on version 2 of the Internet Group Management Protocol (IGMP) for IPv4, and MLD version 2 is based on version 3 of the IGMP for IPv4. IPv6 multicast for Cisco IOS software uses both MLD version 2 and MLD version 1. MLD version 2 is fully backward-compatible with MLD version 1 (described in RFC 2710). Hosts that support only MLD version 1 will interoperate with a switch running MLD version 2. Mixed LANs with both MLD version 1 and MLD version 2 hosts are likewise supported.
- PIM-SM is used between switches so that they can track which multicast packets to forward to each other and to their directly connected LANs.
- PIM in Source Specific Multicast (PIM-SSM) is similar to PIM-SM with the additional ability to report interest in receiving packets from specific source addresses (or from all but the specific source addresses) to an IP multicast address.

MLD Access Group

The MLD access group provides receiver access control in Cisco IOS IPv6 multicast switches. This feature limits the list of groups a receiver can join, and it allows or denies sources used to join SSM channels.

Explicit Tracking of Receivers

The explicit tracking feature allows a switch to track the behavior of the hosts within its IPv6 network. This feature also enables the fast leave mechanism to be used with MLD version 2 host reports.

Protocol Independent Multicast

Protocol Independent Multicast (PIM) is used between switches so that they can track which multicast packets to forward to each other and to their directly connected LANs. PIM works independently of the unicast routing protocol to perform send or receive multicast route updates like other protocols. Regardless of which unicast routing protocols are being used in the LAN to populate the unicast routing table, Cisco IOS PIM uses the existing unicast table content to perform the Reverse Path Forwarding (RPF) check instead of building and maintaining its own separate routing table.

You can configure IPv6 multicast to use either PIM-SM or PIM-SSM operation, or you can use both PIM-SM and PIM-SSM together in your network.

PIM-Sparse Mode

IPv6 multicast provides support for intradomain multicast routing using PIM-SM. PIM-SM uses unicast routing to provide reverse-path information for multicast tree building, but it is not dependent on any particular unicast routing protocol.

PIM-SM is used in a multicast network when relatively few switches are involved in each multicast and these switches do not forward multicast packets for a group, unless there is an explicit request for the traffic. PIM-SM distributes information about active sources by forwarding data packets on the shared tree. PIM-SM initially uses shared trees, which requires the use of an RP.

Requests are accomplished via PIM joins, which are sent hop by hop toward the root node of the tree. The root node of a tree in PIM-SM is the RP in the case of a shared tree or the first-hop switch that is directly connected to the multicast source in the case of a shortest path tree (SPT). The RP keeps track of multicast groups and the hosts that send multicast packets are registered with the RP by that host's first-hop switch.

As a PIM join travels up the tree, switches along the path set up multicast forwarding state so that the requested multicast traffic will be forwarded back down the tree. When multicast traffic is no longer needed, a switch sends a PIM prune up the tree toward the root node to prune (or remove) the unnecessary traffic. As this PIM prune travels hop by hop up the tree, each switch updates its forwarding state appropriately. Ultimately, the forwarding state associated with a multicast group or source is removed.

A multicast data sender sends data destined for a multicast group. The designated switch (DR) of the sender takes those data packets, unicast-encapsulates them, and sends them directly to the RP. The RP receives these encapsulated data packets, de-encapsulates them, and forwards them onto the shared tree. The packets then follow the (*, G) multicast tree state in the switches on the RP tree, being replicated wherever the RP tree branches, and eventually reaching all the receivers for that multicast group. The process of encapsulating data packets to the RP is called registering, and the encapsulation packets are called PIM register packets.

IPv6 BSR: Configure RP Mapping

PIM switches in a domain must be able to map each multicast group to the correct RP address. The BSR protocol for PIM-SM provides a dynamic, adaptive mechanism to distribute group-to-RP mapping information rapidly throughout a domain. With the IPv6 BSR feature, if an RP becomes unreachable, it will be detected and the mapping tables will be modified so that the unreachable RP is no longer used, and the new tables will be rapidly distributed throughout the domain.

Every PIM-SM multicast group needs to be associated with the IP or IPv6 address of an RP. When a new multicast sender starts sending, its local DR will encapsulate these data packets in a PIM register message and send them to the RP for that multicast group. When a new multicast receiver joins, its local DR will send a PIM join message to the RP for that multicast group. When any PIM switch sends a (*, G) join message, the PIM switch needs to know which is the next switch toward the RP so that G (Group) can send a message to that switch. Also, when a PIM switch is forwarding data packets using (*, G) state, the PIM switch needs to know which is the correct incoming interface for packets destined for G, because it needs to reject any packets that arrive on other interfaces.

A small set of switches from a domain are configured as candidate bootstrap switches (C-BSRs) and a single BSR is selected for that domain. A set of switches within a domain are also configured as candidate RPs (C-RPs); typically, these switches are the same switches that are configured as C-BSRs. Candidate RPs periodically unicast candidate-RP-advertisement (C-RP-Adv) messages to the BSR of that domain, advertising their willingness to be an RP. A C-RP-Adv message includes the address of the advertising C-RP, and an optional list of group addresses and mask length fields, indicating the group prefixes for which the candidacy is advertised. The BSR then includes a set of these C-RPs, along with their corresponding group prefixes, in bootstrap messages (BSMs) it periodically originates. BSMs are distributed hop-by-hop throughout the domain.

Bidirectional BSR support allows bidirectional RPs to be advertised in C-RP messages and bidirectional ranges in the BSM. All switches in a system must be able to use the bidirectional range in the BSM; otherwise, the bidirectional RP feature will not function.

PIM-Source Specific Multicast

PIM-SSM is the routing protocol that supports the implementation of SSM and is derived from PIM-SM. However, unlike PIM-SM where data from all multicast sources are sent when there is a PIM join, the SSM feature forwards datagram traffic to receivers from only those multicast sources that the receivers have explicitly joined, thus optimizing bandwidth utilization and denying unwanted Internet broadcast traffic. Further, instead

of the use of RP and shared trees, SSM uses information found on source addresses for a multicast group. This information is provided by receivers through the source addresses relayed to the last-hop switches by MLD membership reports, resulting in shortest-path trees directly to the sources.

In SSM, delivery of datagrams is based on (S, G) channels. Traffic for one (S, G) channel consists of datagrams with an IPv6 unicast source address S and the multicast group address G as the IPv6 destination address. Systems will receive this traffic by becoming members of the (S, G) channel. Signaling is not required, but receivers must subscribe or unsubscribe to (S, G) channels to receive or not receive traffic from specific sources.

MLD version 2 is required for SSM to operate. MLD allows the host to provide source information. Before SSM can run with MLD, SSM must be supported in the Cisco IOS IPv6 switch, the host where the application is running, and the application itself.

Routable Address Hello Option

When an IPv6 interior gateway protocol is used to build the unicast routing table, the procedure to detect the upstream switch address assumes the address of a PIM neighbor is always same as the address of the next-hop switch, as long as they refer to the same switch. However, it may not be the case when a switch has multiple addresses on a link.

Two typical situations can lead to this situation for IPv6. The first situation can occur when the unicast routing table is not built by an IPv6 interior gateway protocol such as multicast BGP. The second situation occurs when the address of an RP shares a subnet prefix with downstream switches (note that the RP switch address has to be domain-wide and therefore cannot be a link-local address).

The routable address hello option allows the PIM protocol to avoid such situations by adding a PIM hello message option that includes all the addresses on the interface on which the PIM hello message is advertised. When a PIM switch finds an upstream switch for some address, the result of RPF calculation is compared with the addresses in this option, in addition to the PIM neighbor's address itself. Because this option includes all the possible addresses of a PIM switch on that link, it always includes the RPF calculation result if it refers to the PIM switch supporting this option.

Because of size restrictions on PIM messages and the requirement that a routable address hello option fits within a single PIM hello message, a limit of 16 addresses can be configured on the interface.

PIM IPv6 Stub Routing

The PIM stub routing feature reduces resource usage by moving routed traffic closer to the end user.

In a network using PIM stub routing, the only allowable route for IPv6 traffic to the user is through a switch that is configured with PIM stub routing. PIM passive interfaces are connected to Layer 2 access domains, such as VLANs, or to interfaces that are connected to other Layer 2 devices. Only directly connected multicast receivers and sources are allowed in the Layer 2 access domains. The PIM passive interfaces do not send or process any received PIM control packets.

When using PIM stub routing, you should configure the distribution and remote routers to use IPv6 multicast routing and configure only the switch as a PIM stub router. The switch does not route transit traffic between distribution routers. You also need to configure a routed uplink port on the switch. The switch uplink port cannot be used with SVIs.

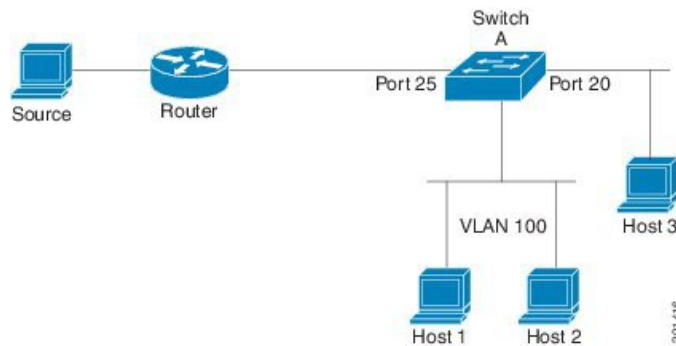
You must also configure EIGRP stub routing when configuring PIM stub routing on the switch.

The redundant PIM stub router topology is not supported. The redundant topology exists when there is more than one PIM router forwarding multicast traffic to a single access domain. PIM messages are blocked, and the PIM assert and designated router election mechanisms are not supported on the PIM passive interfaces.

Only the non-redundant access router topology is supported by the PIM stub feature. By using a non-redundant topology, the PIM passive interface assumes that it is the only interface and designated router on that access domain.

In the figure shown below, Switch A routed uplink port 25 is connected to the router and PIM stub routing is enabled on the VLAN 100 interfaces and on Host 3. This configuration allows the directly connected hosts to receive traffic from multicast source.

Figure 20: PIM Stub Router Configuration



Static Mroutes

IPv6 static mroutes behave much in the same way as IPv4 static mroutes used to influence the RPF check. IPv6 static mroutes share the same database as IPv6 static routes and are implemented by extending static route support for RPF checks. Static mroutes support equal-cost multipath mroutes, and they also support unicast-only static routes.

MRIB

The Multicast Routing Information Base (MRIB) is a protocol-independent repository of multicast routing entries instantiated by multicast routing protocols (routing clients). Its main function is to provide independence between routing protocols and the Multicast Forwarding Information Base (MFIB). It also acts as a coordination and communication point among its clients.

Routing clients use the services provided by the MRIB to instantiate routing entries and retrieve changes made to routing entries by other clients. Besides routing clients, MRIB also has forwarding clients (MFIB instances) and special clients such as MLD. MFIB retrieves its forwarding entries from MRIB and notifies the MRIB of any events related to packet reception. These notifications can either be explicitly requested by routing clients or spontaneously generated by the MFIB.

Another important function of the MRIB is to allow for the coordination of multiple routing clients in establishing multicast connectivity within the same multicast session. MRIB also allows for the coordination between MLD and routing protocols.

MFIB

The MFIB is a platform-independent and routing-protocol-independent library for IPv6 software. Its main purpose is to provide a Cisco IOS platform with an interface with which to read the IPv6 multicast forwarding table and notifications when the forwarding table changes. The information provided by the MFIB has clearly

defined forwarding semantics and is designed to make it easy for the platform to translate to its specific hardware or software forwarding mechanisms.

When routing or topology changes occur in the network, the IPv6 routing table is updated, and those changes are reflected in the MFIB. The MFIB maintains next-hop address information based on the information in the IPv6 routing table. Because there is a one-to-one correlation between MFIB entries and routing table entries, the MFIB contains all known routes and eliminates the need for route cache maintenance that is associated with switching paths such as fast switching and optimum switching.

Distributed MFIB



Note Distributed MFIB has its significance only in a stacked environment where the active switch distributes the MFIB information to the other stack's member switches. In the following section the line cards are nothing but the member switches in the stack.

Distributed MFIB (dMFIB) is used to switch multicast IPv6 packets on distributed platforms. dMFIB may also contain platform-specific information on replication across line cards. The basic MFIB routines that implement the core of the forwarding logic are common to all forwarding environments.

dMFIB implements the following functions:

- Distributes a copy of the MFIB to the line cards.
- Relays data-driven protocol events generated in the line cards to PIM.
- Provides an MFIB platform application program interface (API) to propagate MFIB changes to platform-specific code responsible for programming the hardware acceleration engine. This API also includes entry points to switch a packet in software (necessary if the packet is triggering a data-driven event) and to upload traffic statistics to the software.
- Provides hooks to allow clients residing on the RP to read traffic statistics on demand. (dMFIB does not periodically upload these statistics to the RP.)

The combination of dMFIB and MRIB subsystems also allows the switch to have a "customized" copy of the MFIB database in each line card and to transport MFIB-related platform-specific information from the RP to the line cards.

IPv6 Multicast Process Switching and Fast Switching

A unified MFIB is used to provide both fast switching and process switching support for PIM-SM and PIM-SSM in IPv6 multicast. In process switching, the Route Processor must examine, rewrite, and forward each packet. The packet is first received and copied into the system memory. The switch then looks up the Layer 3 network address in the routing table. The Layer 2 frame is then rewritten with the next-hop destination address and sent to the outgoing interface. The RP also computes the cyclic redundancy check (CRC). This switching method is the least scalable method for switching IPv6 packets.

IPv6 multicast fast switching allows switches to provide better packet forwarding performance than process switching. Information conventionally stored in a route cache is stored in several data structures for IPv6 multicast switching. The data structures provide optimized lookup for efficient packet forwarding.

In IPv6 multicast forwarding, the first packet is fast-switched if the PIM protocol logic allows it. In IPv6 multicast fast switching, the MAC encapsulation header is precomputed. IPv6 multicast fast switching uses

the MFIB to make IPv6 destination prefix-based switching decisions. In addition to the MFIB, IPv6 multicast fast switching uses adjacency tables to prepend Layer 2 addressing information. The adjacency table maintains Layer 2 next-hop addresses for all MFIB entries.

The adjacency table is populated as adjacencies are discovered. Each time an adjacency entry is created (such as through ARP), a link-layer header for that adjacent node is precomputed and stored in the adjacency table. Once a route is determined, it points to a next hop and corresponding adjacency entry. It is subsequently used for encapsulation during switching of packets.

A route might have several paths to a destination prefix, such as when a switch is configured for simultaneous load balancing and redundancy. For each resolved path, a pointer is added for the adjacency corresponding to the next-hop interface for that path. This mechanism is used for load balancing across several paths.

Implementing IPv6 Multicast

Enabling IPv6 Multicast Routing

To enable IPv6 multicast routing, perform this procedure:

Procedure

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	configure terminal	Enter global configuration mode.
Step 3	ipv6 multicast-routing Example: Device(config)# ipv6 multicast-routing	Enables multicast routing on all IPv6-enabled interfaces and enables multicast forwarding for PIM and MLD on all enabled interfaces of the switch.
Step 4	copy running-config startup-config	(Optional) Save your entries in the configuration file.

Customizing and Verifying the MLD Protocol

Customizing and Verifying MLD on an Interface

To customize and verify MLD on an interface, perform this procedure:

Procedure

	Command or Action	Purpose
Step 1	enable Example:	Enables privileged EXEC mode. Enter your password if prompted.

	Command or Action	Purpose
	Device> enable	
Step 2	configure terminal	Enters global configuration mode.
Step 3	interface <i>type number</i> Example: Device(config)# interface GigabitEthernet 1/0/1	Specifies an interface type and number, and places the switch in interface configuration mode.
Step 4	ipv6 mld join-group [<i>group-address</i>] [include exclude] { <i>source-address</i> source-list [<i>acl</i>]} Example: Device(config-if)# ipv6 mld join-group FF04::10	Configures MLD reporting for a specified group and source.
Step 5	ipv6 mld access-group <i>access-list-name</i> Example: Device(config-if)# ipv6 access-list acc-grp-1	Allows the user to perform IPv6 multicast receiver access control.
Step 6	ipv6 mld static-group [<i>group-address</i>] [include exclude] { <i>source-address</i> source-list [<i>acl</i>]} Example: Device(config-if)# ipv6 mld static-group ff04::10 include 100::1	Statically forwards traffic for the multicast group onto a specified interface and cause the interface to behave as if a MLD joiner were present on the interface.
Step 7	ipv6 mld query-max-response-time <i>seconds</i> Example: Device(config-if)# ipv6 mld query-timeout 130	Configures the timeout value before the switch takes over as the querier for the interface.
Step 8	exit Example: Device(config-if)# exit	Enter this command twice to exit interface configuration mode and enter privileged EXEC mode.
Step 9	show ipv6 mld groups [link-local] [<i>group-name</i> <i>group-address</i>] [<i>interface-type interface-number</i>] [detail explicit] Example: Device# show ipv6 mld groups GigabitEthernet 1/0/1	Displays the multicast groups that are directly connected to the switch and that were learned through MLD.
Step 10	show ipv6 mld groups summary Example:	Displays the number of (*, G) and (S, G) membership reports present in the MLD cache.

	Command or Action	Purpose
	Device# <code>show ipv6 mld groups summary</code>	
Step 11	show ipv6 mld interface [<i>type number</i>] Example: Device# <code>show ipv6 mld interface GigabitEthernet 1/0/1</code>	Displays multicast-related information about an interface.
Step 12	debug ipv6 mld [<i>group-name</i> <i>group-address</i> <i>interface-type</i>] Example: Device# <code>debug ipv6 mld</code>	Enables debugging on MLD protocol activity.
Step 13	debug ipv6 mld explicit [<i>group-name</i> <i>group-address</i>] Example: Device# <code>debug ipv6 mld explicit</code>	Displays information related to the explicit tracking of hosts.
Step 14	copy running-config startup-config	(Optional) Save your entries in the configuration file.

Implementing MLD Group Limits

Per-interface and global MLD limits operate independently of each other. Both per-interface and global MLD limits can be configured on the same switch. The number of MLD limits, globally or per interface, is not configured by default; the limits must be configured by the user. A membership report that exceeds either the per-interface or the global state limit is ignored.

Configuring Explicit Tracking of Receivers to Track Host Behavior

The explicit tracking feature allows a switch to track the behavior of the hosts within its IPv6 network and enables the fast leave mechanism to be used with MLD version 2 host reports.

To configuring explicit tracking of receivers to track host behavior, perform this procedure:

Procedure

	Command or Action	Purpose
Step 1	enable Example: Device> <code>enable</code>	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	configure terminal	Enter global configuration mode.
Step 3	interface <i>type number</i> Example:	Specifies an interface type and number, and places the switch in interface configuration mode.

	Command or Action	Purpose
	Device(config)# interface GigabitEthernet 1/0/1	
Step 4	ipv6 mld explicit-tracking <i>access-list-name</i> Example: Device(config-if)# ipv6 mld explicit-tracking list1	Enables explicit tracking of hosts.
Step 5	copy running-config startup-config	(Optional) Save your entries in the configuration file.

Resetting the MLD Traffic Counters

To reset the MLD traffic counters, perform this procedure:

Procedure

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	clear ipv6 mld traffic Example: Device# clear ipv6 mld traffic	Resets all MLD traffic counters.
Step 4	show ipv6 mld traffic Example: Device# show ipv6 mld traffic	Displays the MLD traffic counters.
Step 5	copy running-config startup-config	(Optional) Save your entries in the configuration file.

Clearing the MLD Interface Counters

To clearing the MLD interface counters, perform this procedure

Procedure

	Command or Action	Purpose
Step 1	enable	Enables privileged EXEC mode.

	Command or Action	Purpose
	Example: Device> enable	Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	clear ipv6 mld counters <i>interface-type</i> Example: Device# clear ipv6 mld counters Ethernet1/0	Clears the MLD interface counters.
Step 4	copy running-config startup-config	(Optional) Save your entries in the configuration file.

Configuring PIM

This section explains how to configure PIM.

Configuring PIM-SM and Displaying PIM-SM Information for a Group Range

To configuring PIM-SM and view PIM-SM information for a group range, perform this procedure:

Procedure

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	ipv6 pim rp-address <i>ipv6-address[group-access-list]</i> Example: Device(config)# ipv6 pim rp-address 2001:DB8::01:800:200E:8C6C acc-grp-1	Configures the address of a PIM RP for a particular group range.
Step 4	exit Example:	Exits global configuration mode, and returns the switch to privileged EXEC mode.

	Command or Action	Purpose
	Device(config)# exit	
Step 5	show ipv6 pim interface [state-on] [state-off] [type-number] Example: Device# show ipv6 pim interface	Displays information about interfaces configured for PIM.
Step 6	show ipv6 pim group-map [group-name group-address] [group-range group-mask] [info-source {bsr default embedded-rp static}] Example: Device# show ipv6 pim group-map	Displays an IPv6 multicast group mapping table.
Step 7	show ipv6 pim neighbor [detail] [interface-type interface-number count] Example: Device# show ipv6 pim neighbor	Displays the PIM neighbors discovered by the Cisco IOS software.
Step 8	show ipv6 pim range-list [config] [rp-address rp-name] Example: Device# show ipv6 pim range-list	Displays information about IPv6 multicast range lists.
Step 9	show ipv6 pim tunnel [interface-type interface-number] Example: Device# show ipv6 pim tunnel	Displays information about the PIM register encapsulation and de-encapsulation tunnels on an interface.
Step 10	debug ipv6 pim [group-name group-address interface interface-type bsr group mvpn neighbor] Example: Device# debug ipv6 pim	Enables debugging on PIM protocol activity.
Step 11	copy running-config startup-config	(Optional) Save your entries in the configuration file.

Configuring PIM Options

To configure PIM options, perform this procedure:

Procedure

	Command or Action	Purpose
Step 1	enable	Enables privileged EXEC mode.

	Command or Action	Purpose
	Example: Device> enable	Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	ipv6 pim spt-threshold infinity [group-list access-list-name] Example: Device(config)# ipv6 pim spt-threshold infinity group-list acc-grp-1	Configures when a PIM leaf switch joins the SPT for the specified groups.
Step 4	ipv6 pim accept-register {list access-list route-map map-name} Example: Device(config)# ipv6 pim accept-register route-map reg-filter	Accepts or rejects registers at the RP.
Step 5	interface type number Example: Device(config)# interface GigabitEthernet 1/0/1	Specifies an interface type and number, and places the switch in interface configuration mode.
Step 6	ipv6 pim dr-priority value Example: Device(config-if)# ipv6 pim dr-priority 3	Configures the DR priority on a PIM switch.
Step 7	ipv6 pim hello-interval seconds Example: Device(config-if)# ipv6 pim hello-interval 45	Configures the frequency of PIM hello messages on an interface.
Step 8	ipv6 pim join-prune-interval seconds Example: Device(config-if)# ipv6 pim join-prune-interval 75	Configures periodic join and prune announcement intervals for a specified interface.
Step 9	exit Example:	Enter this command twice to exit interface configuration mode and enter privileged EXEC mode.

	Command or Action	Purpose
	Device(config-if)# exit	
Step 10	ipv6 pim join-prune statistic [<i>interface-type</i>] Example: Device(config-if)# show ipv6 pim join-prune statistic	Displays the average join-prune aggregation for the most recently aggregated packets for each interface.
Step 11	copy running-config startup-config	(Optional) Save your entries in the configuration file.

Resetting the PIM Traffic Counters

If PIM malfunctions or in order to verify that the expected number of PIM packets are received and sent, the user can clear PIM traffic counters. Once the traffic counters are cleared, the user can enter the `show ipv6 pim traffic` command to verify that PIM is functioning correctly and that PIM packets are being received and sent correctly.

To resetting the PIM traffic counters, perform this procedure:

Procedure

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	clear ipv6 pim traffic Example: Device# clear ipv6 pim traffic	Resets the PIM traffic counters.
Step 4	show ipv6 pim traffic Example: Device# show ipv6 pim traffic	Displays the PIM traffic counters.
Step 5	copy running-config startup-config	(Optional) Save your entries in the configuration file.

Clearing the PIM Topology Table to Reset the MRIB Connection

No configuration is necessary to use the MRIB. However, users may in certain situations want to clear the PIM topology table in order to reset the MRIB connection and verify MRIB information.

To clear the PIM topology table to reset the MRIB connection, perform this procedure:

Procedure

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	clear ipv6 pim topology [<i>group-name</i> <i>group-address</i>] Example: Device# clear ipv6 pim topology FF04::10	Clears the PIM topology table.
Step 4	show ipv6 mrrib client [filter] [name { <i>client-name</i> <i>client-name</i> : <i>client-id</i> }] Example: Device# show ipv6 mrrib client	Displays multicast-related information about an interface.
Step 5	show ipv6 mrrib route { link-local summary [<i>sourceaddress-or-name</i> *] [<i>groupname-or-address</i> [<i>prefix-length</i>]]] Example: Device# show ipv6 mrrib route	Displays the MRIB route information.
Step 6	show ipv6 pim topology [<i>groupname-or-address</i> [<i>sourceaddress-or-name</i>] link-local route-count [detail]] Example: Device# show ipv6 pim topology	Displays PIM topology table information for a specific group or all groups.
Step 7	debug ipv6 mrrib client Example: Device# debug ipv6 mrrib client	Enables debugging on MRIB client management activity.

	Command or Action	Purpose
Step 8	debug ipv6 mrib io Example: Device# <code>debug ipv6 mrib io</code>	Enables debugging on MRIB I/O events.
Step 9	debug ipv6 mrib proxy Example: Device# <code>debug ipv6 mrib proxy</code>	Enables debugging on MRIB proxy activity between the switch processor and line cards on distributed switch platforms.
Step 10	debug ipv6 mrib route [<i>group-name</i> <i>group-address</i>] Example: Device# <code>debug ipv6 mrib route</code>	Displays information about MRIB routing entry-related activity.
Step 11	debug ipv6 mrib table Example: Device# <code>debug ipv6 mrib table</code>	Enables debugging on MRIB table management activity.
Step 12	copy running-config startup-config	(Optional) Save your entries in the configuration file.

Configuring PIM IPv6 Stub Routing

The PIM Stub routing feature supports multicast routing between the distribution layer and the access layer. It supports two types of PIM interfaces, uplink PIM interfaces, and PIM passive interfaces. A routed interface configured with the PIM passive mode does not pass or forward PIM control traffic, it only passes and forwards MLD traffic.

PIM IPv6 Stub Routing Configuration Guidelines

- Before configuring PIM stub routing, you must have IPv6 multicast routing configured on both the stub router and the central router. You must also have PIM mode (sparse-mode) configured on the uplink interface of the stub router.
- The PIM stub router does not route the transit traffic between the distribution routers. Unicast (EIGRP) stub routing enforces this behavior. You must configure unicast stub routing to assist the PIM stub router behavior. For more information, see the *EIGRP Stub Routing* section.
- Only directly connected multicast (MLD) receivers and sources are allowed in the Layer 2 access domains. The PIM protocol is not supported in access domains.
- The redundant PIM stub router topology is not supported.

Default IPv6 PIM Routing Configuration

This table displays the default IPv6 PIM routing configuration for the Device.

Table 41: Default Multicast Routing Configuration

Feature	Default Setting
Multicast routing	Disabled on all interfaces.
PIM version	Version 2.
PIM mode	No mode is defined.
PIM stub routing	None configured.
PIM RP address	None configured.
PIM domain border	Disabled.
PIM multicast boundary	None.
Candidate BSRs	Disabled.
Candidate RPs	Disabled.
Shortest-path tree threshold rate	0 kb/s.
PIM router query message interval	30 seconds.

Enabling IPV6 PIM Stub Routing

To enable IPV6 PIM stub routing, perform this procedure:

Before you begin

PIM stub routing is disabled in IPv6 by default.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ipv6 multicast pim-passive-enable**
4. **interface** *interface-id*
5. **ipv6 pim**
6. **ipv6 pim** {bsr} | {dr-priority | *value*} | {hello-interval | *seconds*} | {join-prune-interval | *seconds*} | {passive}
7. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable	Enables privileged EXEC mode.

	Command or Action	Purpose
	Example: Device> enable	Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	ipv6 multicast pim-passive-enable Example: Device(config-if)# ipv6 multicast pim-passive-enable	Enables IPv6 Multicast PIM routing on the switch.
Step 4	interface interface-id Example: Device(config)# interface gigabitethernet 9/0/6	Specifies the interface on which you want to enable PIM stub routing, and enters interface configuration mode. The specified interface must be one of the following: <ul style="list-style-type: none"> • A routed port—A physical port that has been configured as a Layer 3 port by entering the no switchport interface configuration command. You will also need to enable IP PIM sparse mode on the interface, and join the interface as a statically connected member to an MLD static group. • An SVI—A VLAN interface created by using the interface vlan vlan-id global configuration command. You will also need to enable IP PIM sparse mode on the VLAN, join the VLAN as a statically connected member to an MLD static group, and then enable MLD snooping on the VLAN, the MLD static group, and physical interface. These interfaces must have IPv6 addresses assigned to them.
Step 5	ipv6 pim Example: Device(config-if)# ipv6 pim	Enables the PIM on the interface.
Step 6	ipv6 pim {bsr} {dr-priority value} {hello-interval seconds} {join-prune-interval seconds} {passive} Example: Device(config-if)# ipv6 pim bsr dr-priority hello-interval join-prune-interval passive	Configures the various PIM stub features on the interface. Enter bsr to configure BSR on a PIM switch Enter dr-priority to configure the DR priority on a PIM switch.

	Command or Action	Purpose
		Enter hello-interval to configure the frequency of PIM hello messages on an interface. Enter join-prune-interval to configure periodic join and prune announcement intervals for a specified interface. Enter passive to configure the PIM in the passive mode.
Step 7	end Example: Device(config-if) # end	Returns to privileged EXEC mode.

Monitoring IPv6 PIM Stub Routing

Table 42: PIM Stub Configuration show Commands

Command	Purpose
show ipv6 pim interface Device# show ipv6 pim interface	Displays the PIM stub that is enabled on each interface.
show ipv6 mld groups Device# show ipv6 mld groups	Displays the interested clients that have joined the specific multicast source group.
show ipv6 mroute Device# show ipv6 mroute	Verifies that the multicast stream forwards from the source to the interested clients.

Configuring a BSR

The tasks included here are described below.

Configuring a BSR and Verifying BSR Information

To configure and verify BSR Information, perform this procedure:

Procedure

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. Enter your password if prompted.

	Command or Action	Purpose
Step 2	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode.
Step 3	ipv6 pim bsr candidate bsr <i>ipv6-address[hash-mask-length] [priority priority-value]</i> Example: Device(config)# <code>ipv6 pim bsr candidate bsr 2001:DB8:3000:3000::42 124 priority 10</code>	Configures a switch to be a candidate BSR.
Step 4	interface type number Example: Device(config)# <code>interface GigabitEthernet 1/0/1</code>	Specifies an interface type and number, and places the switch in interface configuration mode.
Step 5	ipv6 pim bsr border Example: Device(config-if)# <code>ipv6 pim bsr border</code>	Specifies an interface type and number, and places the switch in interface configuration mode.
Step 6	exit Example: Device(config-if)# <code>exit</code>	Enter this command twice to exit interface configuration mode and enter privileged EXEC mode.
Step 7	show ipv6 pim bsr {election rp-cache candidate-rp} Example: Device(config-if)# <code>show ipv6 pim bsr election</code>	Displays information related to PIM BSR protocol processing.
Step 8	copy running-config startup-config	(Optional) Save your entries in the configuration file.

Sending PIM RP Advertisements to the BSR

To sending PIM RP advertisements to the BSR, perform this procedure:

Procedure

	Command or Action	Purpose
Step 1	enable Example: Device> <code>enable</code>	Enables privileged EXEC mode. Enter your password if prompted.

	Command or Action	Purpose
Step 2	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode.
Step 3	ipv6 pim bsr candidate rp <i>ipv6-address</i> [group-list <i>access-list-name</i>] [priority <i>priority-value</i>] [interval <i>seconds</i>] Example: Device(config)# <code>ipv6 pim bsr candidate rp 2001:DB8:3000:3000::42 priority 0</code>	Sends PIM RP advertisements to the BSR.
Step 4	interface <i>type number</i> Example: Device(config)# <code>interface GigabitEthernet 1/0/1</code>	Specifies an interface type and number, and places the switch in interface configuration mode.
Step 5	ipv6 pim bsr border Example: Device(config-if)# <code>ipv6 pim bsr border</code>	Configures a border for all BSMs of any scope on a specified interface.
Step 6	copy running-config startup-config	(Optional) Save your entries in the configuration file.

Configuring BSR for Use Within Scoped Zones

To configure BSR for use within scoped zones, perform this procedure:

Procedure

	Command or Action	Purpose
Step 1	enable Example: Device> <code>enable</code>	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode.
Step 3	ipv6 pim bsr candidate rp <i>ipv6-address</i> [<i>hash-mask-length</i>] [priority <i>priority-value</i>] Example:	Configures a switch to be a candidate BSR.

	Command or Action	Purpose
	Device(config)# ipv6 pim bsr candidate bsr 2001:DB8:1:1:4	
Step 4	ipv6 pim bsr candidate rp <i>ipv6-address</i> [group-list <i>access-list-name</i>] [priority <i>priority-value</i>] [interval <i>seconds</i>] Example: Device(config)# ipv6 pim bsr candidate rp 2001:DB8:1:1:1 group-list list scope 6	Configures the candidate RP to send PIM RP advertisements to the BSR.
Step 5	interface <i>type number</i> Example: Device(config-if)# interface GigabitEthernet 1/0/1	Specifies an interface type and number, and places the switch in interface configuration mode.
Step 6	ipv6 multicast boundary scope <i>scope-value</i> Example: Device(config-if)# ipv6 multicast boundary scope 6	Configures a multicast boundary on the interface for a specified scope.
Step 7	copy running-config startup-config	(Optional) Save your entries in the configuration file.

Configuring BSR Switches to Announce Scope-to-RP Mappings

IPv6 BSR switches can be statically configured to announce scope-to-RP mappings directly instead of learning them from candidate-RP messages. A user might want to configure a BSR switch to announce scope-to-RP mappings so that an RP that does not support BSR is imported into the BSR. Enabling this feature also allows an RP positioned outside the enterprise's BSR domain to be learned by the known remote RP on the local candidate BSR switch.

To configure BSR switches to announce Scope-to-RP mappings, perform this procedure:

Procedure

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.

	Command or Action	Purpose
Step 3	ipv6 pim bsr announced rp <i>ipv6-address</i> [group-list <i>access-list-name</i>] [priority <i>priority-value</i>] Example: <pre>Device(config)# ipv6 pim bsr announced rp 2001:DB8:3000:3000::42 priority 0</pre>	Announces scope-to-RP mappings directly from the BSR for the specified candidate RP.
Step 4	copy running-config startup-config	(Optional) Save your entries in the configuration file.

Configuring SSM Mapping

When the SSM mapping feature is enabled, DNS-based SSM mapping is automatically enabled, which means that the switch will look up the source of a multicast MLD version 1 report from a DNS server.

You can use either DNS-based or static SSM mapping, depending on your switch configuration. If you choose to use static SSM mapping, you can configure multiple static SSM mappings. If multiple static SSM mappings are configured, the source addresses of all matching access lists will be used.



Note To use DNS-based SSM mapping, the switch needs to find at least one correctly configured DNS server, to which the switch may be directly attached.

To configuring SSM mapping, perform this procedure:

Procedure

	Command or Action	Purpose
Step 1	enable Example: <pre>Device> enable</pre>	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	configure terminal Example: <pre>Device# configure terminal</pre>	Enters global configuration mode.
Step 3	ipv6 mld ssm-map enable Example: <pre>Device(config)# ipv6 mld ssm-map enable</pre>	Enables the SSM mapping feature for groups in the configured SSM range.
Step 4	no ipv6 mld ssm-map query dns Example:	Disables DNS-based SSM mapping.

	Command or Action	Purpose
	Device(config)# no ipv6 mld ssm-map query dns	
Step 5	ipv6 mld ssm-map static <i>access-list source-address</i> Example: Device(config-if)# ipv6 mld ssm-map static SSM_MAP_ACL_2 2001:DB8:1::1	Configures static SSM mappings.
Step 6	exit Example: Device(config-if)# exit	Exits global configuration mode, and returns the switch to privileged EXEC mode.
Step 7	show ipv6 mld ssm-map [<i>source-address</i>] Example: Device(config-if)# show ipv6 mld ssm-map	Displays SSM mapping information.
Step 8	copy running-config startup-config	(Optional) Save your entries in the configuration file.

Configuring Static Mroutes

Static multicast routes (mroutes) in IPv6 can be implemented as an extension of IPv6 static routes. You can configure your switch to use a static route for unicast routing only, to use a static multicast route for multicast RPF selection only, or to use a static route for both unicast routing and multicast RPF selection.

To configure static mroutes, perform this procedure:

Procedure

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	ipv6 route { <i>ipv6-prefix / prefix-length ipv6-address</i> <i>interface-type interface-number ipv6-address</i> } [<i>administrative-distance</i>] [<i>administrative-multicast-distance</i>] <i>unicast</i> <i>multicast</i>] [tag tag] Example:	Establishes static IPv6 routes. The example shows a static route used for both unicast routing and multicast RPF selection.

	Command or Action	Purpose
	Device(config)# ipv6 route 2001:DB8::/64 6::6 100	
Step 4	exit Example: Device# exit	Exits global configuration mode, and returns the switch to privileged EXEC mode.
Step 5	show ipv6 mroute [link-local [group-name group-address [source-address source-name]] [summary] [count] Example: Device# show ipv6 mroute ff07::1	Displays the contents of the IPv6 multicast routing table.
Step 6	show ipv6 mroute [link-local group-name group-address] active [kbps] Example: Device(config-if)# show ipv6 mroute active	Displays the active multicast streams on the switch.
Step 7	show ipv6 rpf [ipv6-prefix] Example: Device(config-if)# show ipv6 rpf 2001::1:1:2	Checks RPF information for a given unicast host address and prefix.
Step 8	copy running-config startup-config	(Optional) Save your entries in the configuration file.

Using MFIB in IPv6 Multicast

Multicast forwarding is automatically enabled when IPv6 multicast routing is enabled.

Verifying MFIB Operation in IPv6 Multicast

To verify MFIB operation in IPv6 multicast

Procedure

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	show ipv6 mfib [link-local verbose group-address-name ipv6-prefix / prefix-length source-address-name active count interface status summary]	Displays the forwarding entries and interfaces in the IPv6 MFIB.

	Command or Action	Purpose
	Example: Device# <code>show ipv6 mfib</code>	
Step 3	show ipv6 mfib [<i>link-local</i> <i>group-name</i> <i>group-address</i>] active [<i>kpbs</i>] Example: Device# <code>show ipv6 mfib active</code>	Displays the rate at which active sources are sending to multicast groups.
Step 4	show ipv6 mfib [<i>all</i> <i>linkscope</i> <i>group-name</i> <i>group-address</i> [<i>source-name</i> <i>source-address</i>]] count Example: Device# <code>show ipv6 mfib ff07::1</code>	Displays the contents of the IPv6 multicast routing table.
Step 5	show ipv6 mfib interface Example: Device# <code>show ipv6 mfib interface</code>	Displays information about IPv6 multicast-enabled interfaces and their forwarding status.
Step 6	show ipv6 mfib status Example: Device# <code>show ipv6 mfib status</code>	Displays general MFIB configuration and operational status.
Step 7	show ipv6 mfib summary Example: Device# <code>show ipv6 mfib summary</code>	Displays summary information about the number of IPv6 MFIB entries and interfaces.
Step 8	debug ipv6 mfib [<i>group-name</i> <i>group-address</i>] [adjacency db fs init interface mrrib [detail] nat pak platform ppr ps signal table] Example: Device# <code>debug ipv6 mfib FF04::10 pak</code>	Enables debugging output on the IPv6 MFIB.

Resetting MFIB Traffic Counters

To reset MFIB traffic counters, perform this procedure:

Procedure

	Command or Action	Purpose
Step 1	enable Example:	Enables privileged EXEC mode. Enter your password if prompted.

	Command or Action	Purpose
	Device> enable	
Step 2	clear ipv6 mfib counters [<i>group-name</i> group-address [<i>source-address</i> <i>source-name</i>]] Example: Device# clear ipv6 mfib counters FF04::10	Resets all active MFIB traffic counters.



CHAPTER 24

Configuring IPv6 ACL

- [Information About Configuring IPv6 ACLs, on page 365](#)
- [Configuring IPv6 ACLs, on page 367](#)
- [Configuration Examples for IPv6 ACL, on page 373](#)

Information About Configuring IPv6 ACLs

You can filter IP version 6 (IPv6) traffic by creating IPv6 access control lists (ACLs) and applying them to interfaces similarly to the way that you create and apply IP version 4 (IPv4) named ACLs. You can also create and apply input router ACLs to filter Layer 3 management traffic.



Note To use IPv6, you must configure the dual IPv4 and IPv6 Switch Database Management (SDM) template on the switch. You select the template by entering the `sdm prefer {default }` global configuration command.

Understanding IPv6 ACLs

A switch image supports two types of IPv6 ACLs:

- IPv6 router ACLs - Supported on inbound or outbound traffic on Layer 3 interfaces, which can be routed ports, switch virtual interfaces (SVIs), or Layer 3 EtherChannels. Applied to only IPv6 packets that are routed.
- IPv6 port ACLs - Supported on inbound traffic on Layer 2 interfaces only. Applied to all IPv6 packets entering the interface.



Note If you configure unsupported IPv6 ACLs, an error message appears and the configuration does not take effect.

The switch does not support VLAN ACLs (VLAN maps) for IPv6 traffic.

You can apply both IPv4 and IPv6 ACLs to an interface.

As with IPv4 ACLs, IPv6 port ACLs take precedence over router ACLs:

- When an input router ACL and input port ACL exist in an SVI, packets received on ports to which a port ACL is applied are filtered by the port ACL. Routed IP packets received on other ports are filtered by the router ACL. Other packets are not filtered.
- When an output router ACL and input port ACL exist in an SVI, packets received on the ports to which a port ACL is applied are filtered by the port ACL. Outgoing routed IPv6 packets are filtered by the router ACL. Other packets are not filtered.



Note If any port ACL (IPv4, IPv6, or MAC) is applied to an interface, that port ACL is used to filter packets, and any router ACLs attached to the SVI of the port VLAN are ignored.

Supported ACL Features

IPv6 ACLs on the switch have these characteristics:

- Fragmented frames (the fragments keyword as in IPv4) are supported.
- The same statistics supported in IPv4 are supported for IPv6 ACLs.
- If the switch runs out of TCAM space, packets associated with the ACL label are forwarded to the CPU, and the ACLs are applied in software.
- Routed or bridged packets with hop-by-hop options have IPv6 ACLs applied in software.
- Logging is supported for router ACLs, but not for port ACLs.

IPv6 ACL Limitations

With IPv4, you can configure standard and extended numbered IP ACLs, named IP ACLs, and MAC ACLs. IPv6 supports only named ACLs.

The switch supports most Cisco IOS-supported IPv6 ACLs with some exceptions:

- IPv6 source and destination addresses-ACL matching is supported only on prefixes from /0 to /64 and host addresses (/128) that are in the extended universal identifier (EUI)-64 format. The switch supports only these host addresses with no loss of information:
 - aggregatable global unicast addresses
 - link local addresses
- The switch does not support matching on these keywords: **flowlabel**, **routing header**, and **undetermined-transport**.
- The switch does not support reflexive ACLs (the **reflect** keyword).
- This release supports only port ACLs and router ACLs for IPv6; it does not support VLAN ACLs (VLAN maps).
- The switch does not apply MAC-based ACLs on IPv6 frames.
- You cannot apply IPv6 port ACLs to Layer 2 EtherChannels.
- The switch does not support output port ACLs.

- Output router ACLs and input port ACLs for IPv6 are supported only on . Switches support only control plane (incoming) IPv6 ACLs.
- When configuring an ACL, there is no restriction on keywords entered in the ACL, regardless of whether or not they are supported on the platform. When you apply the ACL to an interface that requires hardware forwarding (physical ports or SVIs), the switch checks to determine whether or not the ACL can be supported on the interface. If not, attaching the ACL is rejected.
- If an ACL is applied to an interface and you attempt to add an access control entry (ACE) with an unsupported keyword, the switch does not allow the ACE to be added to the ACL that is currently attached to the interface.

Configuring IPv6 ACLs

To filter IPv6 traffic, you perform these steps:

Before you begin

Before configuring IPv6 ACLs, you must select one of the dual IPv4 and IPv6 SDM templates.

Procedure

	Command or Action	Purpose
Step 1	Create an IPv6 ACL, and enter IPv6 access list configuration mode.	
Step 2	Configure the IPv6 ACL to block (deny) or pass (permit) traffic.	
Step 3	Apply the IPv6 ACL to an interface. For router ACLs, you must also configure an IPv6 address on the Layer 3 interface to which the ACL is applied.	

Default IPv6 ACL Configuration

There are no IPv6 ACLs configured or applied.

Interaction with Other Features and Switches

- If an IPv6 router ACL is configured to deny a packet, the packet is not routed. A copy of the packet is sent to the Internet Control Message Protocol (ICMP) queue to generate an ICMP unreachable message for the frame.
- If a bridged frame is to be dropped due to a port ACL, the frame is not bridged.
- You can create both IPv4 and IPv6 ACLs on a switch or switch stack, and you can apply both IPv4 and IPv6 ACLs to the same interface. Each ACL must have a unique name; an error message appears if you try to use a name that is already configured.

You use different commands to create IPv4 and IPv6 ACLs and to attach IPv4 or IPv6 ACLs to the same Layer 2 or Layer 3 interface. If you use the wrong command to attach an ACL (for example, an IPv4 command to attach an IPv6 ACL), you receive an error message.

- You cannot use MAC ACLs to filter IPv6 frames. MAC ACLs can only filter non-IP frames.
- If the hardware memory is full, for any additional configured ACLs, packets are dropped to the CPU, and the ACLs are applied in software. When the hardware is full a message is printed to the console indicating the ACL has been unloaded and the packets will be dropped on the interface.



Note Only packets of the same type as the ACL that could not be added (ipv4, ipv6, MAC) will be dropped on the interface.

Creating an IPv6 ACL

Procedure

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	ipv6 access-list <i>acl_name</i> Example: Device# ipv6 access-list access-list-name	Use a name to define an IPv6 access list and enter IPv6 access-list configuration mode.
Step 4	{deny permit} protocol Example: <pre>{deny permit} protocol {source-ipv6-prefix/prefix-length any host source-ipv6-address} [operator [port-number]][destination-ipv6-prefix/prefix-length any host destination-ipv6-address} [operator [port-number]][dscp value] [fragments][log] [log-input] [routing][sequence value] [time-range name]</pre>	Enter deny or permit to specify whether to deny or permit the packet if conditions are matched. These are the conditions: <ul style="list-style-type: none"> • For protocol, enter the name or number of an Internet protocol: ahp, esp, icmp, ipv6, pcp, stcp, tcp, or udp, or an integer in the range 0 to 255 representing an IPv6 protocol number. • The source-ipv6-prefix/prefix-length or destination-ipv6-prefix/ prefix-length is the source or destination IPv6 network or class of networks for which to set deny or permit conditions, specified in hexadecimal and using 16-bit values between colons (see RFC 2373).

	Command or Action	Purpose
		<ul style="list-style-type: none"> • Enter any as an abbreviation for the IPv6 prefix ::/0. • For host source-ipv6-address or destination-ipv6-address, enter the source or destination IPv6 host address for which to set deny or permit conditions, specified in hexadecimal using 16-bit values between colons. • (Optional) For operator, specify an operand that compares the source or destination ports of the specified protocol. Operands are lt (less than), gt (greater than), eq (equal), neq (not equal), and range. <p>If the operator follows the source-ipv6-prefix/prefix-length argument, it must match the source port. If the operator follows the destination-ipv6- prefix/prefix-length argument, it must match the destination port.</p> <ul style="list-style-type: none"> • (Optional) The port-number is a decimal number from 0 to 65535 or the name of a TCP or UDP port. You can use TCP port names only when filtering TCP. You can use UDP port names only when filtering UDP. • (Optional) Enter dscp value to match a differentiated services code point value against the traffic class value in the Traffic Class field of each IPv6 packet header. The acceptable range is from 0 to 63. • (Optional) Enter fragments to check noninitial fragments. This keyword is visible only if the protocol is ipv6. • (Optional) Enter log to cause an logging message to be sent to the console about the packet that matches the entry. Enter log-input to include the input interface in the log entry. Logging is supported only for router ACLs. • (Optional) Enter routing to specify that IPv6 packets be routed. • (Optional) Enter sequence value to specify the sequence number for the access list statement. The acceptable range is from 1 to 4294967295 • (Optional) Enter time-range name to specify the time range that applies to the deny or permit statement.
Step 5	{deny permit} tcp Example:	(Optional) Define a TCP access list and the access conditions.

	Command or Action	Purpose
	<pre>{deny permit} tcp {source-ipv6-prefix/prefix-length any hostsource-ipv6-address} [operator [port-number]]{destination-ipv6-prefix/prefix-length any hostdestination-ipv6-address} [operator [port-number]][ack] [dscp value][established] [fin] [log][log-input] [neq {port protocol}] [psh] [range{port protocol}] [rst][routing] [sequence value] [syn] [time-range name][urg]</pre>	<p>Enter <code>tcp</code> for Transmission Control Protocol. The parameters are the same as those described in Step 3, with these additional optional parameters:</p> <ul style="list-style-type: none"> • <code>ack</code>—Acknowledgment bit set. • <code>established</code>—An established connection. A match occurs if the TCP datagram has the ACK or RST bits set. • <code>fin</code>—Finished bit set; no more data from sender. • <code>neq {port protocol}</code>—Matches only packets that are not on a given port number. • <code>psh</code>—Push function bit set. • <code>range {port protocol}</code>—Matches only packets in the port number range. • <code>rst</code>—Reset bit set. • <code>syn</code>—Synchronize bit set. • <code>urg</code>—Urgent pointer bit set.
Step 6	<p>{deny permit} udp</p> <p>Example:</p> <pre>{deny permit} udp {source-ipv6-prefix/prefix-length any hostsource-ipv6-address} [operator [port-number]]{destination-ipv6-prefix/prefix-length any hostdestination-ipv6-address} [operator [port-number]][dscp value] [log][log-input] [neq {port protocol}] [range {port protocol}] [routing][sequence value][time-range name]</pre>	<p>(Optional) Define a UDP access list and the access conditions.</p> <p>Enter <code>udp</code> for the User Datagram Protocol. The UDP parameters are the same as those described for TCP, except that the operator <code>[port]</code> port number or name must be a UDP port number or name, and the <code>established</code> parameter is not valid for UDP.</p>
Step 7	<p>{deny permit} icmp</p> <p>Example:</p> <pre>{deny permit} icmp {source-ipv6-prefix/prefix-length any hostsource-ipv6-address} [operator [port-number]] {destination-ipv6-prefix/prefix-length any hostdestination-ipv6-address} [operator [port-number]][icmp-type [icmp-code] icmp-message] [dscpvalue] [log] [log-input] [routing] [sequence value][time-range name]</pre>	<p>(Optional) Define an ICMP access list and the access conditions.</p> <p>Enter <code>icmp</code> for Internet Control Message Protocol. The ICMP parameters are the same as those described for most IP protocols in Step 3a, with the addition of the ICMP message type and code parameters. These optional keywords have these meanings:</p> <ul style="list-style-type: none"> • <code>icmp-type</code>—Enter to filter by ICMP message type, a number from 0 to 255. • <code>icmp-code</code>—Enter to filter ICMP packets that are filtered by the ICMP message code type, a number from 0 to 255.

	Command or Action	Purpose
		<ul style="list-style-type: none"> icmp-message—Enter to filter ICMP packets by the ICMP message type name or the ICMP message type and code name. To see a list of ICMP message type names and code names, use the ? key or see command reference for this release.
Step 8	end Example: Device(config)# end	Returns to privileged EXEC mode. Alternatively, you can also press Ctrl-Z to exit global configuration mode.
Step 9	show ipv6 access-list Example: show ipv6 access-list	Verify the access list configuration.
Step 10	copy running-config startup-config Example: copy running-config startup-config	(Optional) Save your entries in the configuration file.

Applying an IPv6 to an Interface

This section describes how to apply IPv6 ACLs to network interfaces. You can apply an IPv6 ACL to outbound or inbound traffic on layer 2 and Layer 3 interfaces. You can apply IPv6 ACLs only to inbound management traffic on Layer 3 interfaces.

To control access to an interface, perform this procedure:

Procedure

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	interface interface_id Example: Device# interface interface-id	Identifies a Layer 2 interface (for port ACLs) or Layer 3 Switch Virtual interface (for router ACLs) on which to apply an access list, and enters interface configuration mode.

	Command or Action	Purpose
Step 4	no switchport Example: Device# no switchport	Changes the interface from Layer 2 mode (the default) to Layer 3 mode (only if applying a router ACL).
Step 5	ipv6 address ipv6_address Example: Device# ipv6 address ipv6-address	Configures an IPv6 address on a Layer 3 interface (for router ACLs). Note This command is not required on Layer 2 interfaces or if the interface has already been configured with an explicit IPv6 address.
Step 6	ipv6 traffic-filter acl_name Example: Device# ipv6 traffic-filter access-list-name {in out}	Applies the access list to incoming or outgoing traffic on the interface.
Step 7	end Example: Device(config)# end	Returns to privileged EXEC mode. Alternatively, you can also press Ctrl-Z to exit global configuration mode.
Step 8	show running-config interface tenGigabitEthernet 1/0/3 Example: Device# show running-config interface tenGigabitEthernet 1/0/3 Building configuration, Current configuration : 98 bytes ! interface TenGigabitEthernet1/0/3 switchport mode trunk ipv6 traffic-filter MyFilter out end	Shows the configuration summary.
Step 9	copy running-config startup-config Example: copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Displaying IPv6 ACLs

To display IPv6 ACLs, perform this procedure:

Procedure

	Command or Action	Purpose
Step 1	enable Example:	Enables privileged EXEC mode. Enter your password if prompted.

	Command or Action	Purpose
	Device> enable	
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	show access-list Example: Device# show access-lists	Displays all access lists configured on the device
Step 4	show ipv6 access-list <i>acl_name</i> Example: Device# show ipv6 access-list [access-list-name]	Displays all configured IPv6 access list or the access list specified by name.

Configuration Examples for IPv6 ACL

Example: Creating an IPv6 ACL

This example configures the IPv6 access list named CISCO. The first deny entry in the list denies all packets that have a destination TCP port number greater than 5000. The second deny entry denies packets that have a source UDP port number less than 5000. The second deny also logs all matches to the console. The first permit entry in the list permits all ICMP packets. The second permit entry in the list permits all other traffic. The second permit entry is necessary because an implicit deny -all condition is at the end of each IPv6 access list.



Note Logging is supported only on Layer 3 interfaces.

```
Device(config)# ipv6 access-list CISCO
Device(config-ipv6-acl)# deny tcp any any gt 5000
Device (config-ipv6-acl)# deny ::/0 lt 5000 ::/0 log
Device(config-ipv6-acl)# permit icmp any any
Device(config-ipv6-acl)# permit any any
```

Example: Applying IPv6 ACLs

This example shows how to apply the access list Cisco to outbound traffic on a Layer 3 interface.

```
Device(config-if)# no switchport
Device(config-if)# ipv6 address 2001::/64 eui-64
Device(config-if)# ipv6 traffic-filter CISCO out
```

Example: Displaying IPv6 ACLs

This is an example of the output from the **show access-lists** privileged EXEC command. The output shows all access lists that are configured on the switch or switch stack.

```
Device #show access-lists
Extended IP access list hello
10 permit ip any any
IPv6 access list ipv6
permit ipv6 any any sequence 10
```

This is an example of the output from the **show ipv6 access-lists** privileged EXEC command. The output shows only IPv6 access lists configured on the switch or switch stack.

```
Device# show ipv6 access-list
IPv6 access list inbound
permit tcp any any eq bgp (8 matches) sequence 10
permit tcp any any eq telnet (15 matches) sequence 20
permit udp any any sequence 30

IPv6 access list outbound
deny udp any any sequence 10
deny tcp any any eq telnet sequence 20
```




PART **V**

Layer 2/3

- [Configuring Spanning Tree Protocol, on page 377](#)
- [Configuring Multiple Spanning-Tree Protocol, on page 403](#)
- [Configuring Optional Spanning-Tree Features, on page 445](#)
- [Configuring EtherChannels, on page 477](#)
- [Configuring Link-State Tracking, on page 515](#)
- [Configuring Flex Links and the MAC Address-Table Move Update Feature, on page 521](#)
- [Configuring UniDirectional Link Detection, on page 539](#)
- [Configuring Resilient Ethernet Protocol, on page 547](#)
- [Configuring the PPPoE Intermediate Agent, on page 565](#)



CHAPTER 25

Configuring Spanning Tree Protocol

This chapter describes how to configure the Spanning Tree Protocol (STP) on port-based VLANs on the Catalyst devices. The device can use either the per-VLAN spanning-tree plus (PVST+) protocol based on the IEEE 802.1D standard and Cisco proprietary extensions, or the rapid per-VLAN spanning-tree plus (rapid-PVST+) protocol based on the IEEE 802.1w standard. A switch stack appears as a single spanning-tree node to the rest of the network, and all stack members use the same bridge ID.

- [Restrictions for STP, on page 377](#)
- [Information About Spanning Tree Protocol, on page 377](#)
- [How to Configure Spanning-Tree Features, on page 389](#)
- [Monitoring Spanning-Tree Status, on page 401](#)
- [Additional References for Spanning-Tree Protocol, on page 401](#)
- [Feature Information for STP, on page 402](#)

Restrictions for STP

- An attempt to configure a device as the root device fails if the value necessary to be the root device is less than 1.
- If your network consists of devices that support and do not support the extended system ID, it is unlikely that the device with the extended system ID support will become the root device. The extended system ID increases the device priority value every time the VLAN number is greater than the priority of the connected devices running older software.
- The root device for each spanning-tree instance should be a backbone or distribution device. Do not configure an access device as the spanning-tree primary root.

Information About Spanning Tree Protocol

Spanning Tree Protocol

Spanning Tree Protocol (STP) is a Layer 2 link management protocol that provides path redundancy while preventing loops in the network. For a Layer 2 Ethernet network to function properly, only one active path can exist between any two stations. Multiple active paths among end stations cause loops in the network. If a loop exists in the network, end stations might receive duplicate messages. Devices might also learn end-station

MAC addresses on multiple Layer 2 interfaces. These conditions result in an unstable network. Spanning-tree operation is transparent to end stations, which cannot detect whether they are connected to a single LAN segment or a switched LAN of multiple segments.

The STP uses a spanning-tree algorithm to select one device of a redundantly connected network as the root of the spanning tree. The algorithm calculates the best loop-free path through a switched Layer 2 network by assigning a role to each port based on the role of the port in the active topology:

- Root—A forwarding port elected for the spanning-tree topology
- Designated—A forwarding port elected for every switched LAN segment
- Alternate—A blocked port providing an alternate path to the root bridge in the spanning tree
- Backup—A blocked port in a loopback configuration

The device that has *all* of its ports as the designated role or as the backup role is the root device. The device that has at least *one* of its ports in the designated role is called the designated device.

Spanning tree forces redundant data paths into a standby (blocked) state. If a network segment in the spanning tree fails and a redundant path exists, the spanning-tree algorithm recalculates the spanning-tree topology and activates the standby path. Devices send and receive spanning-tree frames, called bridge protocol data units (BPDUs), at regular intervals. The devices do not forward these frames but use them to construct a loop-free path. BPDUs contain information about the sending device and its ports, including device and MAC addresses, device priority, port priority, and path cost. Spanning tree uses this information to elect the root device and root port for the switched network and the root port and designated port for each switched segment.

When two ports on a device are part of a loop, the spanning-tree and path cost settings control which port is put in the forwarding state and which is put in the blocking state. The spanning-tree port priority value represents the location of a port in the network topology and how well it is located to pass traffic. The path cost value represents the media speed.



Note By default, the device sends keepalive messages (to ensure the connection is up) only on interfaces that do not have small form-factor pluggable (SFP) modules. You can change the default for an interface by entering the `[no] keepalive` interface configuration command with no keywords.

Spanning-Tree Topology and BPDUs

The stable, active spanning-tree topology of a switched network is controlled by these elements:

- The unique bridge ID (device priority and MAC address) associated with each VLAN on each device. In a device stack, all devices use the same bridge ID for a given spanning-tree instance.
- The spanning-tree path cost to the root device.
- The port identifier (port priority and MAC address) associated with each Layer 2 interface.

When the devices in a network are powered up, each functions as the root device. Each device sends a configuration BPDU through all of its ports. The BPDUs communicate and compute the spanning-tree topology. Each configuration BPDU contains this information:

- The unique bridge ID of the device that the sending device identifies as the root device
- The spanning-tree path cost to the root

- The bridge ID of the sending device
- Message age
- The identifier of the sending interface
- Values for the hello, forward delay, and max-age protocol timers

When a device receives a configuration BPDU that contains *superior* information (lower bridge ID, lower path cost, and so forth), it stores the information for that port. If this BPDU is received on the root port of the device, the device also forwards it with an updated message to all attached LANs for which it is the designated device.

If a device receives a configuration BPDU that contains *inferior* information to that currently stored for that port, it discards the BPDU. If the device is a designated device for the LAN from which the inferior BPDU was received, it sends that LAN a BPDU containing the up-to-date information stored for that port. In this way, inferior information is discarded, and superior information is propagated on the network.

A BPDU exchange results in these actions:

- One device in the network is elected as the root device (the logical center of the spanning-tree topology in a switched network). See the figure following the bullets.
For each VLAN, the device with the highest device priority (the lowest numerical priority value) is elected as the root device. If all devices are configured with the default priority (32768), the device with the lowest MAC address in the VLAN becomes the root device. The device priority value occupies the most significant bits of the bridge ID, as shown in the following figure.
- A root port is selected for each device (except the root device). This port provides the best path (lowest cost) when the device forwards packets to the root device.
- Only one outgoing port on the stack root device is selected as the root port. The remaining devices in the stack become its designated devices (Device 2 and Device 3) as shown in the following figure.
- The shortest distance to the root device is calculated for each device based on the path cost.
- A designated device for each LAN segment is selected. The designated device incurs the lowest path cost when forwarding packets from that LAN to the root device. The port through which the designated device is attached to the LAN is called the designated port.

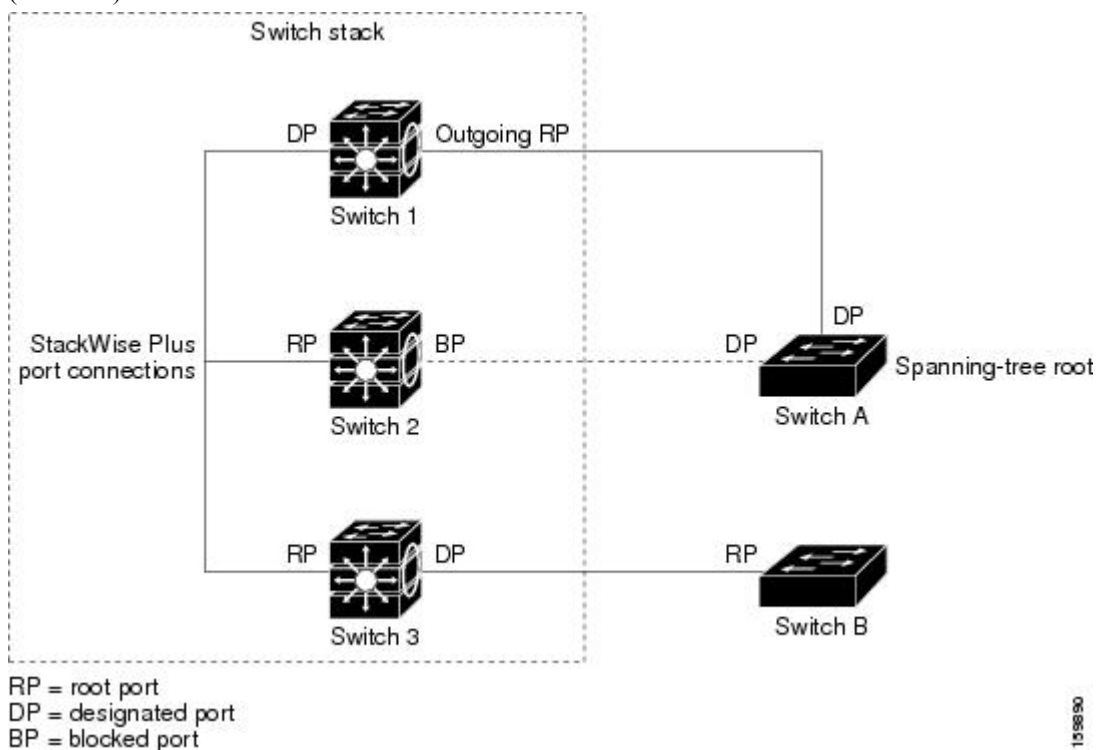


Note If the **logging event spanning tree** command is configured on multiple interfaces and the topology changes, it may result in several logging messages and high CPU utilization. This may cause the switch to drop or delay the processing of STP BPDUs.

To prevent this behavior, remove the **logging event spanning tree** and **logging event status** commands or disable logging to the console.

Figure 21: Spanning-Tree Port States in a Device Stack

One stack member is elected as the stack root device. The stack root device contains the outgoing root port (Device 1).



All paths that are not needed to reach the root device from anywhere in the switched network are placed in the spanning-tree blocking mode.

Bridge ID, Device Priority, and Extended System ID

The IEEE 802.1D standard requires that each device has a unique bridge identifier (bridge ID), which controls the selection of the root device. Because each VLAN is considered as a different *logical bridge* with PVST+ and Rapid PVST+, the same device must have a different bridge ID for each configured VLAN. Each VLAN on the device has a unique 8-byte bridge ID. The 2 most-significant bytes are used for the device priority, and the remaining 6 bytes are derived from the device MAC address.

The device supports the IEEE 802.1t spanning-tree extensions, and some of the bits previously used for the device priority are now used as the VLAN identifier. The result is that fewer MAC addresses are reserved for the device, and a larger range of VLAN IDs can be supported, all while maintaining the uniqueness of the bridge ID.

The 2 bytes previously used for the device priority are reallocated into a 4-bit priority value and a 12-bit extended system ID value equal to the VLAN ID.

Table 43: Device Priority Value and Extended System ID

Priority Value				Extended System ID (Set Equal to the VLAN ID)											
Bit 16	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1
32768	16384	8192	4096	2048	1024	512	256	128	64	32	16	8	4	2	1

Spanning tree uses the extended system ID, the device priority, and the allocated spanning-tree MAC address to make the bridge ID unique for each VLAN. Because the device stack appears as a single device to the rest of the network, all devices in the stack use the same bridge ID for a given spanning tree. If the stack's active switch fails, the stack members recalculate their bridge IDs of all running spanning trees based on the new MAC address of the new stack's active switch.

Support for the extended system ID affects how you manually configure the root device, the secondary root device, and the device priority of a VLAN. For example, when you change the device priority value, you change the probability that the device will be elected as the root device. Configuring a higher value decreases the probability; a lower value increases the probability.

If any root device for the specified VLAN has a device priority lower than 24576, the device sets its own priority for the specified VLAN to 4096 less than the lowest device priority. 4096 is the value of the least-significant bit of a 4-bit device priority value as shown in the table.

Port Priority Versus Path Cost

If a loop occurs, spanning tree uses port priority when selecting an interface to put into the forwarding state. You can assign higher priority values (lower numerical values) to interfaces that you want selected first and lower priority values (higher numerical values) that you want selected last. If all interfaces have the same priority value, spanning tree puts the interface with the lowest interface number in the forwarding state and blocks the other interfaces.

The spanning-tree path cost default value is derived from the media speed of an interface. If a loop occurs, spanning tree uses cost when selecting an interface to put in the forwarding state. You can assign lower cost values to interfaces that you want selected first and higher cost values that you want selected last. If all interfaces have the same cost value, spanning tree puts the interface with the lowest interface number in the forwarding state and blocks the other interfaces.

If your device is a member of a device stack, you must assign lower cost values to interfaces that you want selected first and higher cost values that you want selected last instead of adjusting its port priority. For details, see Related Topics.

Spanning-Tree Interface States

Propagation delays can occur when protocol information passes through a switched LAN. As a result, topology changes can take place at different times and at different places in a switched network. When an interface transitions directly from nonparticipation in the spanning-tree topology to the forwarding state, it can create temporary data loops. Interfaces must wait for new topology information to propagate through the switched LAN before starting to forward frames. They must allow the frame lifetime to expire for forwarded frames that have used the old topology.

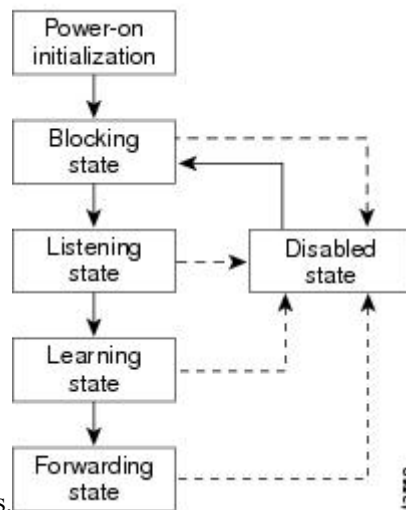
Each Layer 2 interface on a device using spanning tree exists in one of these states:

- Blocking—The interface does not participate in frame forwarding.
- Listening—The first transitional state after the blocking state when the spanning tree decides that the interface should participate in frame forwarding.
- Learning—The interface prepares to participate in frame forwarding.
- Forwarding—The interface forwards frames.
- Disabled—The interface is not participating in spanning tree because of a shutdown port, no link on the port, or no spanning-tree instance running on the port.

An interface moves through these states:

- From initialization to blocking
- From blocking to listening or to disabled
- From listening to learning or to disabled
- From learning to forwarding or to disabled
- From forwarding to disabled

Figure 22: Spanning-Tree Interface States



An interface moves through the states.

When you power up the device, spanning tree is enabled by default, and every interface in the device, VLAN, or network goes through the blocking state and the transitory states of listening and learning. Spanning tree stabilizes each interface at the forwarding or blocking state.

When the spanning-tree algorithm places a Layer 2 interface in the forwarding state, this process occurs:

1. The interface is in the listening state while spanning tree waits for protocol information to move the interface to the blocking state.
2. While spanning tree waits for the forward-delay timer to expire, it moves the interface to the learning state and resets the forward-delay timer.
3. In the learning state, the interface continues to block frame forwarding as the device learns end-station location information for the forwarding database.

4. When the forward-delay timer expires, spanning tree moves the interface to the forwarding state, where both learning and frame forwarding are enabled.

Blocking State

A Layer 2 interface in the blocking state does not participate in frame forwarding. After initialization, a BPDU is sent to each device interface. A device initially functions as the root until it exchanges BPDUs with other devices. This exchange establishes which device in the network is the root or root device. If there is only one device in the network, no exchange occurs, the forward-delay timer expires, and the interface moves to the listening state. An interface always enters the blocking state after device initialization.

An interface in the blocking state performs these functions:

- Discards frames received on the interface
- Discards frames switched from another interface for forwarding
- Does not learn addresses
- Receives BPDUs

Listening State

The listening state is the first state a Layer 2 interface enters after the blocking state. The interface enters this state when the spanning tree decides that the interface should participate in frame forwarding.

An interface in the listening state performs these functions:

- Discards frames received on the interface
- Discards frames switched from another interface for forwarding
- Does not learn addresses
- Receives BPDUs

Learning State

A Layer 2 interface in the learning state prepares to participate in frame forwarding. The interface enters the learning state from the listening state.

An interface in the learning state performs these functions:

- Discards frames received on the interface
- Discards frames switched from another interface for forwarding
- Learns addresses
- Receives BPDUs

Forwarding State

A Layer 2 interface in the forwarding state forwards frames. The interface enters the forwarding state from the learning state.

An interface in the forwarding state performs these functions:

- Receives and forwards frames received on the interface

- Forwards frames switched from another interface
- Learns addresses
- Receives BPDUs

Disabled State

A Layer 2 interface in the disabled state does not participate in frame forwarding or in the spanning tree. An interface in the disabled state is nonoperational.

A disabled interface performs these functions:

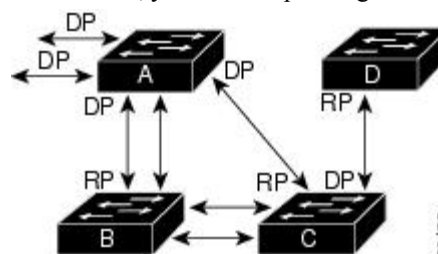
- Discards frames received on the interface
- Discards frames switched from another interface for forwarding
- Does not learn addresses
- Does not receive BPDUs

How a Device or Port Becomes the Root Device or Root Port

If all devices in a network are enabled with default spanning-tree settings, the device with the lowest MAC address becomes the root device.

Figure 23: Spanning-Tree Topology

Device A is elected as the root device because the device priority of all the devices is set to the default (32768) and Device A has the lowest MAC address. However, because of traffic patterns, number of forwarding interfaces, or link types, Device A might not be the ideal root device. By increasing the priority (lowering the numerical value) of the ideal device so that it becomes the root device, you force a spanning-tree recalculation



RP = Root Port
DP = Designated Port

to form a new topology with the ideal device as the root.

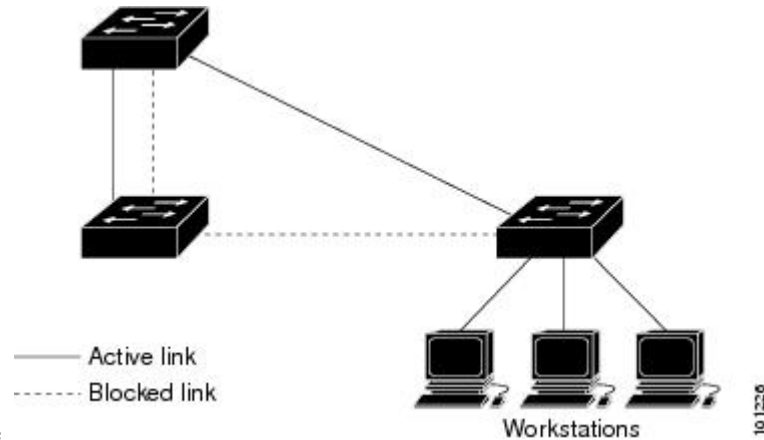
When the spanning-tree topology is calculated based on default parameters, the path between source and destination end stations in a switched network might not be ideal. For instance, connecting higher-speed links to an interface that has a higher number than the root port can cause a root-port change. The goal is to make the fastest link the root port.

For example, assume that one port on Device B is a Gigabit Ethernet link and that another port on Device B (a 10/100 link) is the root port. Network traffic might be more efficient over the Gigabit Ethernet link. By changing the spanning-tree port priority on the Gigabit Ethernet port to a higher priority (lower numerical value) than the root port, the Gigabit Ethernet port becomes the new root port.

Spanning Tree and Redundant Connectivity

Figure 24: Spanning Tree and Redundant Connectivity

You can create a redundant backbone with spanning tree by connecting two device interfaces to another device or to two different devices. Spanning tree automatically disables one interface but enables it if the other one fails. If one link is high-speed and the other is low-speed, the low-speed link is always disabled. If the speeds are the same, the port priority and port ID are added together, and spanning tree disables the link with the



highest value.

You can also create redundant links between devices by using EtherChannel groups.

Spanning-Tree Address Management

IEEE 802.1D specifies 17 multicast addresses, ranging from 0x00180C2000000 to 0x0180C2000010, to be used by different bridge protocols. These addresses are static addresses that cannot be removed.

Regardless of the spanning-tree state, each device in the stack receives but does not forward packets destined for addresses between 0x0180C2000000 and 0x0180C200000F.

If spanning tree is enabled, the CPU on the device or on each device in the stack receives packets destined for 0x0180C2000000 and 0x0180C2000010. If spanning tree is disabled, the device or each device in the stack forwards those packets as unknown multicast addresses.

Accelerated Aging to Retain Connectivity

The default for aging dynamic addresses is 5 minutes, the default setting of the **mac address-table aging-time** global configuration command. However, a spanning-tree reconfiguration can cause many station locations to change. Because these stations could be unreachable for 5 minutes or more during a reconfiguration, the address-aging time is accelerated so that station addresses can be dropped from the address table and then relearned. The accelerated aging is the same as the forward-delay parameter value (**spanning-tree vlan *vlan-id* forward-time seconds** global configuration command) when the spanning tree reconfigures.

Because each VLAN is a separate spanning-tree instance, the device accelerates aging on a per-VLAN basis. A spanning-tree reconfiguration on one VLAN can cause the dynamic addresses learned on that VLAN to be subject to accelerated aging. Dynamic addresses on other VLANs can be unaffected and remain subject to the aging interval entered for the device.

Spanning-Tree Modes and Protocols

The device supports these spanning-tree modes and protocols:

- **PVST+**—This spanning-tree mode is based on the IEEE 802.1D standard and Cisco proprietary extensions. The PVST+ runs on each VLAN on the device up to the maximum supported, ensuring that each has a loop-free path through the network.

The PVST+ provides Layer 2 load-balancing for the VLAN on which it runs. You can create different logical topologies by using the VLANs on your network to ensure that all of your links are used but that no one link is oversubscribed. Each instance of PVST+ on a VLAN has a single root device. This root device propagates the spanning-tree information associated with that VLAN to all other devices in the network. Because each device has the same information about the network, this process ensures that the network topology is maintained.

- **Rapid PVST+**—This spanning-tree mode is the same as PVST+ except that it uses a rapid convergence based on the IEEE 802.1w standard. Beginning from 15.2(4)E release, the STP default mode is Rapid PVST+. To provide rapid convergence, the Rapid PVST+ immediately deletes dynamically learned MAC address entries on a per-port basis upon receiving a topology change. By contrast, PVST+ uses a short aging time for dynamically learned MAC address entries.

Rapid PVST+ uses the same configuration as PVST+ (except where noted), and the device needs only minimal extra configuration. The benefit of Rapid PVST+ is that you can migrate a large PVST+ install base to Rapid PVST+ without having to learn the complexities of the Multiple Spanning Tree Protocol (MSTP) configuration and without having to re-provision your network. In Rapid PVST+ mode, each VLAN runs its own spanning-tree instance up to the maximum supported.

- **MSTP**—This spanning-tree mode is based on the IEEE 802.1s standard. You can map multiple VLANs to the same spanning-tree instance, which reduces the number of spanning-tree instances required to support a large number of VLANs. The MSTP runs on top of the RSTP (based on IEEE 802.1w), which provides for rapid convergence of the spanning tree by eliminating the forward delay and by quickly transitioning root ports and designated ports to the forwarding state. In a device stack, the cross-stack rapid transition (CSRT) feature performs the same function as RSTP. You cannot run MSTP without RSTP or CSRT.

Supported Spanning-Tree Instances

In PVST+ or Rapid PVST+ mode, the device or device stack supports up to 128 spanning-tree instances.

In MSTP mode, the device or device stack supports up to 65 MST instances. The number of VLANs that can be mapped to a particular MST instance is unlimited.

Spanning-Tree Interoperability and Backward Compatibility

In a mixed MSTP and PVST+ network, the common spanning-tree (CST) root must be inside the MST backbone, and a PVST+ device cannot connect to multiple MST regions.

When a network contains devices running Rapid PVST+ and devices running PVST+, we recommend that the Rapid PVST+ devices and PVST+ devices be configured for different spanning-tree instances. In the Rapid PVST+ spanning-tree instances, the root device must be a Rapid PVST+ device. In the PVST+ instances, the root device must be a PVST+ device. The PVST+ devices should be at the edge of the network.

All stack members run the same version of spanning tree (all PVST+, all Rapid PVST+, or all MSTP).

Table 44: PVST+, MSTP, and Rapid-PVST+ Interoperability and Compatibility

	PVST+	MSTP	Rapid PVST+
PVST+	Yes	Yes (with restrictions)	Yes (reverts to PVST+)

	PVST+	MSTP	Rapid PVST+
MSTP	Yes (with restrictions)	Yes	Yes (reverts to PVST+)
Rapid PVST+	Yes (reverts to PVST+)	Yes (reverts to PVST+)	Yes

STP and IEEE 802.1Q Trunks

The IEEE 802.1Q standard for VLAN trunks imposes some limitations on the spanning-tree strategy for a network. The standard requires only one spanning-tree instance for *all* VLANs allowed on the trunks. However, in a network of Cisco devices connected through IEEE 802.1Q trunks, the devices maintain one spanning-tree instance for *each* VLAN allowed on the trunks.

When you connect a Cisco device to a non-Cisco device through an IEEE 802.1Q trunk, the Cisco device uses PVST+ to provide spanning-tree interoperability. If Rapid PVST+ is enabled, the device uses it instead of PVST+. The device combines the spanning-tree instance of the IEEE 802.1Q VLAN of the trunk with the spanning-tree instance of the non-Cisco IEEE 802.1Q device.

However, all PVST+ or Rapid PVST+ information is maintained by Cisco devices separated by a cloud of non-Cisco IEEE 802.1Q devices. The non-Cisco IEEE 802.1Q cloud separating the Cisco devices is treated as a single trunk link between the devices.

Rapid PVST+ is automatically enabled on IEEE 802.1Q trunks, and no user configuration is required. The external spanning-tree behavior on access ports and Inter-Switch Link (ISL) trunk ports is not affected by PVST+.

VLAN-Bridge Spanning Tree

Cisco VLAN-bridge spanning tree is used with the fallback bridging feature (bridge groups), which forwards non-IP protocols such as DECnet between two or more VLAN bridge domains or routed ports. The VLAN-bridge spanning tree allows the bridge groups to form a spanning tree on top of the individual VLAN spanning trees to prevent loops from forming if there are multiple connections among VLANs. It also prevents the individual spanning trees from the VLANs being bridged from collapsing into a single spanning tree.

To support VLAN-bridge spanning tree, some of the spanning-tree timers are increased. To use the fallback bridging feature, you must have the IP services feature set enabled on your device.

Spanning Tree and Device Stacks

When the device stack is operating in PVST+ or Rapid PVST+ mode:

- A device stack appears as a single spanning-tree node to the rest of the network, and all stack members use the same bridge ID for a given spanning tree. The bridge ID is derived from the MAC address of the active stack.
- When a new device joins the stack, it sets its bridge ID to the active stack bridge ID. If the newly added device has the lowest ID and if the root path cost is the same among all stack members, the newly added device becomes the stack root.
- When a stack member leaves the stack, spanning-tree reconvergence occurs within the stack (and possibly outside the stack). The remaining stack member with the lowest stack port ID becomes the stack root.
- If the active stack fails or leaves the stack, the stack members elect a new active stack, and all stack members change their bridge IDs of the spanning trees to the new active stack bridge ID.

- If the device stack is the spanning-tree root and the active stack fails or leaves the stack, the stack members elect a new active stack, and a spanning-tree reconvergence occurs.
- If a neighboring device external to the device stack fails or is powered down, normal spanning-tree processing occurs. Spanning-tree reconvergence might occur as a result of losing a device in the active topology.
- If a new device external to the device stack is added to the network, normal spanning-tree processing occurs. Spanning-tree reconvergence might occur as a result of adding a device in the network.

Default Spanning-Tree Configuration

Table 45: Default Spanning-Tree Configuration

Feature	Default Setting
Enable state	Enabled on VLAN 1.
Spanning-tree mode	Rapid PVST+ (PVST+ and MSTP are disabled.)
Device priority	32768
Spanning-tree port priority (configurable on a per-interface basis)	128
Spanning-tree port cost (configurable on a per-interface basis)	1000 Mb/s: 4 100 Mb/s: 19 10 Mb/s: 100
Spanning-tree VLAN port priority (configurable on a per-VLAN basis)	128
Spanning-tree VLAN port cost (configurable on a per-VLAN basis)	1000 Mb/s: 4 100 Mb/s: 19 10 Mb/s: 100
Spanning-tree timers	Hello time: 2 seconds Forward-delay time: 15 seconds Maximum-aging time: 20 seconds Transmit hold count: 6 BPDUs



Note Beginning in Cisco IOS Release 15.2(4)E, the default STP mode is Rapid PVST+.

How to Configure Spanning-Tree Features

Changing the Spanning-Tree Mode

The switch supports three spanning-tree modes: per-VLAN spanning tree plus (PVST+), Rapid PVST+, or multiple spanning tree protocol (MSTP). By default, the device runs the Rapid PVST+ protocol.

If you want to enable a mode that is different from the default mode, this procedure is required.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **spanning-tree mode {pvst | mst | rapid-pvst}**
4. **interface *interface-id***
5. **spanning-tree link-type point-to-point**
6. **end**
7. **clear spanning-tree detected-protocols**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	spanning-tree mode {pvst mst rapid-pvst} Example: Device(config)# spanning-tree mode pvst	Configures a spanning-tree mode. <p>All stack members run the same version of spanning tree.</p> <ul style="list-style-type: none"> • Select pvst to enable PVST+. • Select mst to enable MSTP. • Select rapid-pvst to enable rapid PVST+.
Step 4	interface <i>interface-id</i> Example: Device(config)# interface GigabitEthernet1/0/1	Specifies an interface to configure, and enters interface configuration mode. Valid interfaces include physical ports, VLANs, and port channels. The VLAN ID range is 1 to 4094. The port-channel range is 1 to 48.

	Command or Action	Purpose
Step 5	spanning-tree link-type point-to-point Example: Device(config-if)# spanning-tree link-type point-to-point	Specifies that the link type for this port is point-to-point. If you connect this port (local port) to a remote port through a point-to-point link and the local port becomes a designated port, the device negotiates with the remote port and rapidly changes the local port to the forwarding state.
Step 6	end Example: Device(config-if)# end	Returns to privileged EXEC mode.
Step 7	clear spanning-tree detected-protocols Example: Device# clear spanning-tree detected-protocols	If any port on the device is connected to a port on a legacy IEEE 802.1D device, this command restarts the protocol migration process on the entire device. This step is optional if the designated device detects that this device is running rapid PVST+.

Disabling Spanning Tree

Spanning tree is enabled by default on VLAN 1 and on all newly created VLANs up to the spanning-tree limit. Disable spanning tree only if you are sure there are no loops in the network topology.



Caution When spanning tree is disabled and loops are present in the topology, excessive traffic and indefinite packet duplication can drastically reduce network performance.

This procedure is optional.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **no spanning-tree vlan *vlan-id***
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.

	Command or Action	Purpose
Step 2	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode.
Step 3	no spanning-tree vlan <i>vlan-id</i> Example: Device(config)# <code>no spanning-tree vlan 300</code>	For <i>vlan-id</i> , the range is 1 to 4094.
Step 4	end Example: Device(config)# <code>end</code>	Returns to privileged EXEC mode.

Configuring the Root Device

To configure a device as the root for the specified VLAN, use the **spanning-tree vlan *vlan-id* root** global configuration command to modify the device priority from the default value (32768) to a significantly lower value. When you enter this command, the software checks the device priority of the root devices for each VLAN. Because of the extended system ID support, the device sets its own priority for the specified VLAN to 24576 if this value will cause this device to become the root for the specified VLAN.

Use the **diameter** keyword to specify the Layer 2 network diameter (that is, the maximum number of device hops between any two end stations in the Layer 2 network). When you specify the network diameter, the device automatically sets an optimal hello time, forward-delay time, and maximum-age time for a network of that diameter, which can significantly reduce the convergence time. You can use the **hello** keyword to override the automatically calculated hello time.

This procedure is optional.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **spanning-tree vlan *vlan-id* root primary [*diameter net-diameter*]**
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> <code>enable</code>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.

	Command or Action	Purpose
Step 2	configure terminal Example: <pre>Device# configure terminal</pre>	Enters global configuration mode.
Step 3	spanning-tree vlan <i>vlan-id</i> root primary [<i>diameter net-diameter</i>] Example: <pre>Device(config)# spanning-tree vlan 20-24 root primary diameter 4</pre>	Configures a device to become the root for the specified VLAN. <ul style="list-style-type: none"> • For <i>vlan-id</i>, you can specify a single VLAN identified by VLAN ID number, a range of VLANs separated by a hyphen, or a series of VLANs separated by a comma. The range is 1 to 4094. • (Optional) For diameter <i>net-diameter</i>, specify the maximum number of devices between any two end stations. The range is 2 to 7.
Step 4	end Example: <pre>Device(config)# end</pre>	Returns to privileged EXEC mode.

What to do next

After configuring the device as the root device, we recommend that you avoid manually configuring the hello time, forward-delay time, and maximum-age time through the **spanning-tree vlan *vlan-id* hello-time**, **spanning-tree vlan *vlan-id* forward-time**, and the **spanning-tree vlan *vlan-id* max-age** global configuration commands.

Configuring a Secondary Root Device

When you configure a device as the secondary root, the device priority is modified from the default value (32768) to 28672. With this priority, the device is likely to become the root device for the specified VLAN if the primary root device fails. This is assuming that the other network devices use the default device priority of 32768, and therefore, are unlikely to become the root device.

You can execute this command on more than one device to configure multiple backup root devices. Use the same network diameter and hello-time values that you used when you configured the primary root device with the **spanning-tree vlan *vlan-id* root primary** global configuration command.

This procedure is optional.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **spanning-tree vlan *vlan-id* root secondary [*diameter net-diameter*]**
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	spanning-tree vlan <i>vlan-id</i> root secondary [<i>diameter net-diameter</i>] Example: Device(config)# spanning-tree vlan 20-24 root secondary diameter 4	Configures a device to become the secondary root for the specified VLAN. <ul style="list-style-type: none"> • For <i>vlan-id</i>, you can specify a single VLAN identified by VLAN ID number, a range of VLANs separated by a hyphen, or a series of VLANs separated by a comma. The range is 1 to 4094. • (Optional) For diameter <i>net-diameter</i>, specify the maximum number of devices between any two end stations. The range is 2 to 7. Use the same network diameter value that you used when configuring the primary root device.
Step 4	end Example: Device(config)# end	Returns to privileged EXEC mode.

Configuring Port Priority



Note If your device is a member of a device stack, you must use the **spanning-tree [vlan *vlan-id*] cost *cost*** interface configuration command instead of the **spanning-tree [vlan *vlan-id*] port-priority *priority*** interface configuration command to select an interface to put in the forwarding state. Assign lower cost values to interfaces that you want selected first and higher cost values that you want selected last.

This procedure is optional.

SUMMARY STEPS

1. **enable**

2. **configure terminal**
3. **interface** *interface-id*
4. **spanning-tree port-priority** *priority*
5. **spanning-tree vlan** *vlan-id* **port-priority** *priority*
6. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	interface <i>interface-id</i> Example: Device(config)# interface gigabitethernet 1/0/2	Specifies an interface to configure, and enters interface configuration mode. Valid interfaces include physical ports and port-channel logical interfaces (port-channel <i>port-channel-number</i>).
Step 4	spanning-tree port-priority <i>priority</i> Example: Device(config-if)# spanning-tree port-priority 0	Configures the port priority for an interface. For <i>priority</i> , the range is 0 to 240, in increments of 16; the default is 128. Valid values are 0, 16, 32, 48, 64, 80, 96, 112, 128, 144, 160, 176, 192, 208, 224, and 240. All other values are rejected. The lower the number, the higher the priority.
Step 5	spanning-tree vlan <i>vlan-id</i> port-priority <i>priority</i> Example: Device(config-if)# spanning-tree vlan 20-25 port-priority 0	Configures the port priority for a VLAN. <ul style="list-style-type: none"> • For <i>vlan-id</i>, you can specify a single VLAN identified by VLAN ID number, a range of VLANs separated by a hyphen, or a series of VLANs separated by a comma. The range is 1 to 4094. • For <i>priority</i>, the range is 0 to 240, in increments of 16; the default is 128. Valid values are 0, 16, 32, 48, 64, 80, 96, 112, 128, 144, 160, 176, 192, 208, 224, and 240. All other values are rejected. The lower the number, the higher the priority.
Step 6	end Example:	Returns to privileged EXEC mode.

	Command or Action	Purpose
	Device(config-if)# end	

Configuring Path Cost

This procedure is optional.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **spanning-tree cost** *cost*
5. **spanning-tree vlan** *vlan-id cost cost*
6. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none">• Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	interface <i>interface-id</i> Example: Device(config)# interface gigabitethernet 1/0/1	Specifies an interface to configure, and enters interface configuration mode. Valid interfaces include physical ports and port-channel logical interfaces (port-channel <i>port-channel-number</i>).
Step 4	spanning-tree cost <i>cost</i> Example: Device(config-if)# spanning-tree cost 250	Configures the cost for an interface. If a loop occurs, spanning tree uses the path cost when selecting an interface to place into the forwarding state. A lower path cost represents higher-speed transmission. For <i>cost</i> , the range is 1 to 200000000; the default value is derived from the media speed of the interface.
Step 5	spanning-tree vlan <i>vlan-id cost cost</i> Example:	Configures the cost for a VLAN.

	Command or Action	Purpose
	<pre>Device(config-if) # spanning-tree vlan 10,12-15,20 cost 300</pre>	<p>If a loop occurs, spanning tree uses the path cost when selecting an interface to place into the forwarding state. A lower path cost represents higher-speed transmission.</p> <ul style="list-style-type: none"> For <i>vlan-id</i>, you can specify a single VLAN identified by VLAN ID number, a range of VLANs separated by a hyphen, or a series of VLANs separated by a comma. The range is 1 to 4094. For <i>cost</i>, the range is 1 to 200000000; the default value is derived from the media speed of the interface.
Step 6	<p>end</p> <p>Example:</p> <pre>Device(config-if) # end</pre>	Returns to privileged EXEC mode.

The **show spanning-tree interface *interface-id*** privileged EXEC command displays information only for ports that are in a link-up operative state. Otherwise, you can use the **show running-config** privileged EXEC command to confirm the configuration.

Configuring the Device Priority of a VLAN

You can configure the device priority and make it more likely that a standalone device or a device in the stack will be chosen as the root device.



Note Exercise care when using this command. For most situations, we recommend that you use the **spanning-tree vlan *vlan-id* root primary** and the **spanning-tree vlan *vlan-id* root secondary** global configuration commands to modify the device priority.

This procedure is optional.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **spanning-tree vlan *vlan-id* priority *priority***
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p>enable</p> <p>Example:</p>	<p>Enables privileged EXEC mode.</p> <ul style="list-style-type: none"> Enter your password if prompted.

	Command or Action	Purpose
	Device> enable	
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	spanning-tree vlan <i>vlan-id</i> priority <i>priority</i> Example: Device(config)# spanning-tree vlan 20 priority 8192	Configures the device priority of a VLAN. <ul style="list-style-type: none"> For <i>vlan-id</i>, you can specify a single VLAN identified by VLAN ID number, a range of VLANs separated by a hyphen, or a series of VLANs separated by a comma. The range is 1 to 4094. For <i>priority</i>, the range is 0 to 61440 in increments of 4096; the default is 32768. The lower the number, the more likely the device will be chosen as the root device. <p>Valid priority values are 4096, 8192, 12288, 16384, 20480, 24576, 28672, 32768, 36864, 40960, 45056, 49152, 53248, 57344, and 61440. All other values are rejected.</p>
Step 4	end Example: Device(config-if)# end	Returns to privileged EXEC mode.

Configuring the Hello Time

The hello time is the time interval between configuration messages generated and sent by the root device.

This procedure is optional.

SUMMARY STEPS

1. **enable**
2. **spanning-tree vlan *vlan-id* hello-time *seconds***
3. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example:	Enables privileged EXEC mode. <ul style="list-style-type: none"> Enter your password if prompted.

	Command or Action	Purpose
	Device> enable	
Step 2	spanning-tree vlan <i>vlan-id</i> hello-time <i>seconds</i> Example: Device(config)# spanning-tree vlan 20-24 hello-time 3	Configures the hello time of a VLAN. The hello time is the time interval between configuration messages generated and sent by the root device. These messages mean that the device is alive. <ul style="list-style-type: none"> • For <i>vlan-id</i>, you can specify a single VLAN identified by VLAN ID number, a range of VLANs separated by a hyphen, or a series of VLANs separated by a comma. The range is 1 to 4094. • For <i>seconds</i>, the range is 1 to 10; the default is 2.
Step 3	end Example: Device(config-if)# end	Returns to privileged EXEC mode.

Configuring the Forwarding-Delay Time for a VLAN

This procedure is optional.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **spanning-tree vlan *vlan-id* forward-time *seconds***
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.

	Command or Action	Purpose
Step 3	spanning-tree vlan <i>vlan-id</i> forward-time <i>seconds</i> Example: <pre>Device(config)# spanning-tree vlan 20,25 forward-time 18</pre>	Configures the forward time of a VLAN. The forwarding delay is the number of seconds an interface waits before changing from its spanning-tree learning and listening states to the forwarding state. <ul style="list-style-type: none"> For <i>vlan-id</i>, you can specify a single VLAN identified by VLAN ID number, a range of VLANs separated by a hyphen, or a series of VLANs separated by a comma. The range is 1 to 4094. For <i>seconds</i>, the range is 4 to 30; the default is 15.
Step 4	end Example: <pre>Device(config)# end</pre>	Returns to privileged EXEC mode.

Configuring the Maximum-Aging Time for a VLAN

This procedure is optional.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **spanning-tree vlan *vlan-id* max-age *seconds***
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Device> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> Enter your password if prompted.
Step 2	configure terminal Example: <pre>Device# configure terminal</pre>	Enters global configuration mode.
Step 3	spanning-tree vlan <i>vlan-id</i> max-age <i>seconds</i> Example:	Configures the maximum-aging time of a VLAN. The maximum-aging time is the number of seconds a device waits without receiving spanning-tree configuration messages before attempting a reconfiguration.

	Command or Action	Purpose
	Device(config)# spanning-tree vlan 20 max-age 30	<ul style="list-style-type: none"> For <i>vlan-id</i>, you can specify a single VLAN identified by VLAN ID number, a range of VLANs separated by a hyphen, or a series of VLANs separated by a comma. The range is 1 to 4094. For <i>seconds</i>, the range is 6 to 40; the default is 20.
Step 4	end Example: Device(config-if)# end	Returns to privileged EXEC mode.

Configuring the Transmit Hold-Count

You can configure the BPDU burst size by changing the transmit hold count value.



Note Changing this parameter to a higher value can have a significant impact on CPU utilization, especially in Rapid PVST+ mode. Lowering this value can slow down convergence in certain scenarios. We recommend that you maintain the default setting.

This procedure is optional.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **spanning-tree transmit hold-count** *value*
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.

	Command or Action	Purpose
Step 3	spanning-tree transmit hold-count <i>value</i> Example: Device(config)# spanning-tree transmit hold-count 6	Configures the number of BPDUs that can be sent before pausing for 1 second. For <i>value</i> , the range is 1 to 20; the default is 6.
Step 4	end Example: Device(config)# end	Returns to privileged EXEC mode.

Monitoring Spanning-Tree Status

Table 46: Commands for Displaying Spanning-Tree Status

show spanning-tree active	Displays spanning-tree information on active interfaces only.
show spanning-tree detail	Displays a detailed summary of interface information.
show spanning-tree vlan <i>vlan-id</i>	Displays spanning-tree information for the specified VLAN.
show spanning-tree interface <i>interface-id</i>	Displays spanning-tree information for the specified interface.
show spanning-tree interface <i>interface-id</i> portfast	Displays spanning-tree portfast information for the specified interface.
show spanning-tree summary [totals]	Displays a summary of interface states or displays the total lines of the STP state section.

To clear spanning-tree counters, use the **clear spanning-tree** [**interface** *interface-id*] privileged EXEC command.

Additional References for Spanning-Tree Protocol

Related Documents

Related Topic	Document Title
Layer 2 commands	<i>Catalyst 2960-XR Switch Layer 2 Command Reference</i>

Standards and RFCs

Standard/RFC	Title
None	—

MIBs

MIB	MIBs Link
All the supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
<p>The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies.</p> <p>To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds.</p> <p>Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.</p>	http://www.cisco.com/support

Feature Information for STP

Release	Modification
Cisco IOS Release 15.0(2)EX1	This feature was introduced.



CHAPTER 26

Configuring Multiple Spanning-Tree Protocol

- [Prerequisites for MSTP, on page 403](#)
- [Restrictions for MSTP, on page 403](#)
- [Information About MSTP, on page 404](#)
- [How to Configure MSTP Features, on page 421](#)
- [Examples, on page 439](#)
- [Additional References for MSTP, on page 443](#)
- [Feature Information for MSTP, on page 444](#)

Prerequisites for MSTP

- For two or more devices to be in the same multiple spanning tree (MST) region, they must have the same VLAN-to-instance map, the same configuration revision number, and the same name.
- For two or more stacked switches to be in the same MST region, they must have the same VLAN-to-instance map, the same configuration revision number, and the same name.
- For load-balancing across redundant paths in the network to work, all VLAN-to-instance mapping assignments must match; otherwise, all traffic flows on a single link. You can achieve load-balancing across a device stack by manually configuring the path cost.
- For load-balancing between a per-VLAN spanning tree plus (PVST+) and an MST cloud or between a rapid-PVST+ and an MST cloud to work, all MST boundary ports must be forwarding. MST boundary ports are forwarding when the root of the internal spanning tree (IST) of the MST cloud is the root of the common spanning tree (CST). If the MST cloud consists of multiple MST regions, one of the MST regions must contain the CST root, and all of the other MST regions must have a better path to the root contained within the MST cloud than a path through the PVST+ or rapid-PVST+ cloud. You might have to manually configure the devices in the clouds.

Restrictions for MSTP

- The device stack supports up to 65 MST instances. The number of VLANs that can be mapped to a particular MST instance is unlimited.
- PVST+, Rapid PVST+, and MSTP are supported, but only one version can be active at any time. (For example, all VLANs run PVST+, all VLANs run Rapid PVST+, or all VLANs run MSTP.)

- All stack members must run the same version of spanning tree (all PVST+, Rapid PVST+, or MSTP).
- VLAN Trunking Protocol (VTP) propagation of the MST configuration is not supported. However, you can manually configure the MST configuration (region name, revision number, and VLAN-to-instance mapping) on each device within the MST region by using the command-line interface (CLI) or through the Simple Network Management Protocol (SNMP) support.
- Partitioning the network into a large number of regions is not recommended. However, if this situation is unavoidable, we recommend that you partition the switched LAN into smaller LANs interconnected by routers or non-Layer 2 devices.
- A region can have one member or multiple members with the same MST configuration; each member must be capable of processing rapid spanning tree protocol (RSTP) Bridge Protocol Data Units (BPDUs). There is no limit to the number of MST regions in a network, but each region can only support up to 65 spanning-tree instances. You can assign a VLAN to only one spanning-tree instance at a time.
- After configuring a device as the root device, we recommend that you avoid manually configuring the hello time, forward-delay time, and maximum-age time through the **spanning-tree mst hello-time**, **spanning-tree mst forward-time**, and the **spanning-tree mst max-age** global configuration commands.

Table 47: PVST+, MSTP, and Rapid PVST+ Interoperability and Compatibility

	PVST+	MSTP	Rapid PVST+
PVST+	Yes	Yes (with restrictions)	Yes (reverts to PVST+)
MSTP	Yes (with restrictions)	Yes	Yes (reverts to PVST+)
Rapid PVST+	Yes (reverts to PVST+)	Yes (reverts to PVST+)	Yes

Information About MSTP

MSTP Configuration

MSTP, which uses RSTP for rapid convergence, enables multiple VLANs to be grouped into and mapped to the same spanning-tree instance, reducing the number of spanning-tree instances needed to support a large number of VLANs. The MSTP provides for multiple forwarding paths for data traffic, enables load balancing, and reduces the number of spanning-tree instances required to support a large number of VLANs. It improves the fault tolerance of the network because a failure in one instance (forwarding path) does not affect other instances (forwarding paths).



Note The multiple spanning-tree (MST) implementation is based on the IEEE 802.1s standard.

The most common initial deployment of MSTP is in the backbone and distribution layers of a Layer 2 switched network. This deployment provides the highly available network required in a service-provider environment.

When the device is in the MST mode, the RSTP, which is based on IEEE 802.1w, is automatically enabled. The RSTP provides rapid convergence of the spanning tree through explicit handshaking that eliminates the IEEE 802.1D forwarding delay and quickly transitions root ports and designated ports to the forwarding state.

Both MSTP and RSTP improve the spanning-tree operation and maintain backward compatibility with equipment that is based on the (original) IEEE 802.1D spanning tree, with existing Cisco-proprietary Multiple Instance STP (MISTP), and with existing Cisco PVST+ and rapid per-VLAN spanning-tree plus (Rapid PVST+).

A device stack appears as a single spanning-tree node to the rest of the network, and all stack members use the same device ID.

MSTP Configuration Guidelines

- When you enable MST by using the **spanning-tree mode mst** global configuration command, RSTP is automatically enabled.
- For configuration guidelines about UplinkFast, BackboneFast, and cross-stack UplinkFast, see the relevant sections in the Related Topics section.
- When the device is in MST mode, it uses the long path-cost calculation method (32 bits) to compute the path cost values. With the long path-cost calculation method, the following path cost values are supported:

Speed	Path Cost Value
10 Mb/s	2,000,000
100 Mb/s	200,000
1 Gb/s	20,000
10 Gb/s	2,000
100 Gb/s	200

Root Switch

The device maintains a spanning-tree instance for the group of VLANs mapped to it. A device ID, consisting of the device priority and the device MAC address, is associated with each instance. For a group of VLANs, the device with the lowest device ID becomes the root device.

When you configure a device as the root, you modify the device priority from the default value (32768) to a significantly lower value so that the device becomes the root device for the specified spanning-tree instance. When you enter this command, the device checks the device priorities of the root devices. Because of the extended system ID support, the device sets its own priority for the specified instance to 24576 if this value will cause this devices to become the root for the specified spanning-tree instance.

If any root device for the specified instance has a device priority lower than 24576, the device sets its own priority to 4096 less than the lowest device priority. (4096 is the value of the least-significant bit of a 4-bit device priority value. For more information, select "Bridge ID, Device Priority, and Extended System ID" link in Related Topics.

If your network consists of devices that support and do not support the extended system ID, it is unlikely that the device with the extended system ID support will become the root device. The extended system ID increases the device priority value every time the VLAN number is greater than the priority of the connected switches running older software.

The root device for each spanning-tree instance should be a backbone or distribution device. Do not configure an access device as the spanning-tree primary root.

Use the **diameter** keyword, which is available only for MST instance 0, to specify the Layer 2 network diameter (that is, the maximum number of device hops between any two end stations in the Layer 2 network). When you specify the network diameter, the device automatically sets an optimal hello time, forward-delay time, and maximum-age time for a network of that diameter, which can significantly reduce the convergence time. You can use the **hello** keyword to override the automatically calculated hello time.

Multiple Spanning-Tree Regions

For switches to participate in multiple spanning-tree (MST) instances, you must consistently configure the switches with the same MST configuration information. A collection of interconnected switches that have the same MST configuration comprises an MST region.

The MST configuration controls to which MST region each device belongs. The configuration includes the name of the region, the revision number, and the MST VLAN-to-instance assignment map. You configure the device for a region by specifying the MST region configuration on it. You can map VLANs to an MST instance, specify the region name, and set the revision number. For instructions and an example, select the "Specifying the MST Region Configuration and Enabling MSTP" link in Related Topics.

A region can have one or multiple members with the same MST configuration. Each member must be capable of processing RSTP bridge protocol data units (BPDUs). There is no limit to the number of MST regions in a network, but each region can support up to 65 spanning-tree instances. Instances can be identified by any number in the range from 0 to 4094. You can assign a VLAN to only one spanning-tree instance at a time.

IST, CIST, and CST

Unlike PVST+ and Rapid PVST+ in which all the spanning-tree instances are independent, the MSTP establishes and maintains two types of spanning trees:

- An internal spanning tree (IST), which is the spanning tree that runs in an MST region.

Within each MST region, the MSTP maintains multiple spanning-tree instances. Instance 0 is a special instance for a region, known as the internal spanning tree (IST). All other MST instances are numbered from 1 to 4094.

The IST is the only spanning-tree instance that sends and receives BPDUs. All of the other spanning-tree instance information is contained in M-records, which are encapsulated within MSTP BPDUs. Because the MSTP BPDU carries information for all instances, the number of BPDUs that need to be processed to support multiple spanning-tree instances is significantly reduced.

All MST instances within the same region share the same protocol timers, but each MST instance has its own topology parameters, such as root device ID, root path cost, and so forth. By default, all VLANs are assigned to the IST.

An MST instance is local to the region; for example, MST instance 1 in region A is independent of MST instance 1 in region B, even if regions A and B are interconnected.

- A common and internal spanning tree (CIST), which is a collection of the ISTs in each MST region, and the common spanning tree (CST) that interconnects the MST regions and single spanning trees.

The spanning tree computed in a region appears as a subtree in the CST that encompasses the entire switched domain. The CIST is formed by the spanning-tree algorithm running among switches that

support the IEEE 802.1w, IEEE 802.1s, and IEEE 802.1D standards. The CIST inside an MST region is the same as the CST outside a region.

Operations Within an MST Region

The IST connects all the MSTP switches in a region. When the IST converges, the root of the IST becomes the CIST regional root (called the *IST master* before the implementation of the IEEE 802.1s standard). It is the device within the region with the lowest device ID and path cost to the CIST root. The CIST regional root is also the CIST root if there is only one region in the network. If the CIST root is outside the region, one of the MSTP switches at the boundary of the region is selected as the CIST regional root.

When an MSTP device initializes, it sends BPDUs claiming itself as the root of the CIST and the CIST regional root, with both of the path costs to the CIST root and to the CIST regional root set to zero. The device also initializes all of its MST instances and claims to be the root for all of them. If the device receives superior MST root information (lower device ID, lower path cost, and so forth) than currently stored for the port, it relinquishes its claim as the CIST regional root.

During initialization, a region might have many subregions, each with its own CIST regional root. As switches receive superior IST information, they leave their old subregions and join the new subregion that contains the true CIST regional root. All subregions shrink except for the one that contains the true CIST regional root.

For correct operation, all switches in the MST region must agree on the same CIST regional root. Therefore, any two switches in the region only synchronize their port roles for an MST instance if they converge to a common CIST regional root.

Operations Between MST Regions

If there are multiple regions or legacy IEEE 802.1D devices within the network, MSTP establishes and maintains the CST, which includes all MST regions and all legacy STP devices in the network. The MST instances combine with the IST at the boundary of the region to become the CST.

The IST connects all the MSTP devices in the region and appears as a subtree in the CIST that encompasses the entire switched domain. The root of the subtree is the CIST regional root. The MST region appears as a virtual device to adjacent STP devices and MST regions.

Only the CST instance sends and receives BPDUs, and MST instances add their spanning-tree information into the BPDUs to interact with neighboring devices and compute the final spanning-tree topology. Because of this, the spanning-tree parameters related to BPDU transmission (for example, hello time, forward time, max-age, and max-hops) are configured only on the CST instance but affect all MST instances. Parameters related to the spanning-tree topology (for example, device priority, port VLAN cost, and port VLAN priority) can be configured on both the CST instance and the MST instance.

MSTP devices use Version 3 RSTP BPDUs or IEEE 802.1D STP BPDUs to communicate with legacy IEEE 802.1D devices. MSTP devices use MSTP BPDUs to communicate with MSTP devices.

IEEE 802.1s Terminology

Some MST naming conventions used in Cisco's prestandard implementation have been changed to identify some *internal* or *regional* parameters. These parameters are significant only within an MST region, as opposed to external parameters that are relevant to the whole network. Because the CIST is the only spanning-tree instance that spans the whole network, only the CIST parameters require the external rather than the internal or regional qualifiers.

- The CIST root is the root device for the unique instance that spans the whole network, the CIST.

- The CIST external root path cost is the cost to the CIST root. This cost is left unchanged within an MST region. Remember that an MST region looks like a single device for the CIST. The CIST external root path cost is the root path cost calculated between these virtual devices and devices that do not belong to any region.
- The CIST regional root was called the IST master in the prestandard implementation. If the CIST root is in the region, the CIST regional root is the CIST root. Otherwise, the CIST regional root is the closest device to the CIST root in the region. The CIST regional root acts as a root device for the IST.
- The CIST internal root path cost is the cost to the CIST regional root in a region. This cost is only relevant to the IST, instance 0.

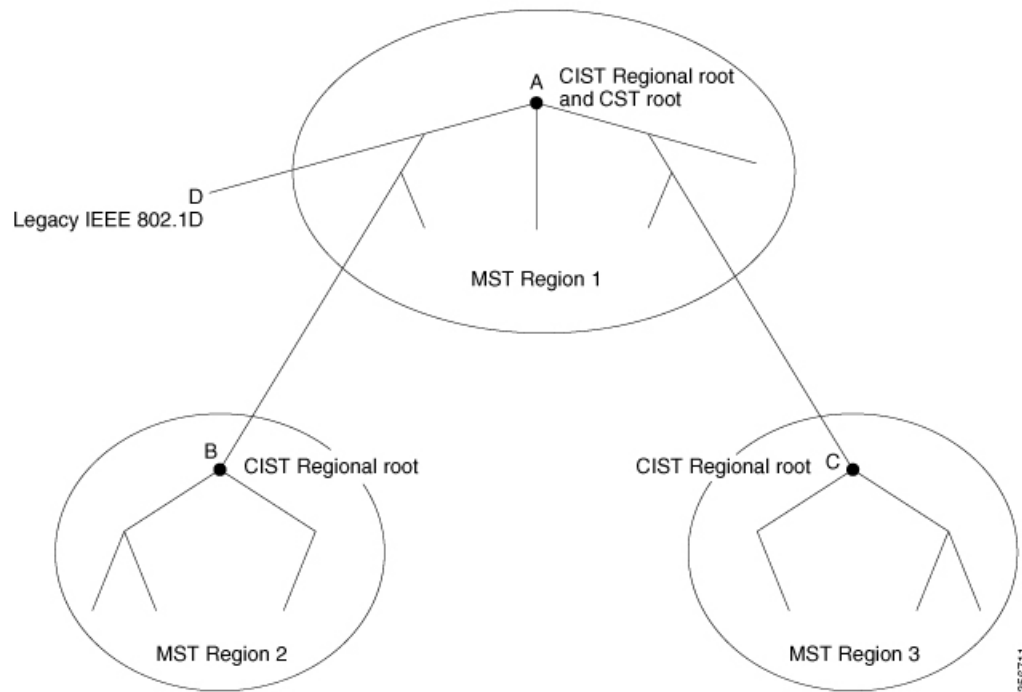
Table 48: Prestandard and Standard Terminology

IEEE Standard	Cisco Prestandard	Cisco Standard
CIST regional root	IST master	CIST regional root
CIST internal root path cost	IST master path cost	CIST internal path cost
CIST external root path cost	Root path cost	Root path cost
MSTI regional root	Instance root	Instance root
MSTI internal root path cost	Root path cost	Root path cost

Illustration of MST Regions

This figure displays three MST regions and a legacy IEEE 802.1D device (D). The CIST regional root for region 1 (A) is also the CIST root. The CIST regional root for region 2 (B) and the CIST regional root for region 3 (C) are the roots for their respective subtrees within the CIST. The RSTP runs in all regions.

Figure 25: MST Regions, CIST Regional Root, and CST Root



Hop Count

The IST and MST instances do not use the message-age and maximum-age information in the configuration BPDU to compute the spanning-tree topology. Instead, they use the path cost to the root and a hop-count mechanism similar to the IP time-to-live (TTL) mechanism.

By using the **spanning-tree mst max-hops** global configuration command, you can configure the maximum hops inside the region and apply it to the IST and all MST instances in that region. The hop count achieves the same result as the message-age information (triggers a reconfiguration). The root device of the instance always sends a BPDU (or M-record) with a cost of 0 and the hop count set to the maximum value. When a device receives this BPDU, it decrements the received remaining hop count by one and propagates this value as the remaining hop count in the BPDUs it generates. When the count reaches zero, the device discards the BPDU and ages the information held for the port.

The message-age and maximum-age information in the RSTP portion of the BPDU remain the same throughout the region, and the same values are propagated by the region designated ports at the boundary.

Boundary Ports

In the Cisco prestandard implementation, a boundary port connects an MST region to a single spanning-tree region running RSTP, to a single spanning-tree region running PVST+ or rapid PVST+, or to another MST region with a different MST configuration. A boundary port also connects to a LAN, the designated device of which is either a single spanning-tree device or a device with a different MST configuration.

There is no definition of a boundary port in the IEEE 802.1s standard. The IEEE 802.1Q-2002 standard identifies two kinds of messages that a port can receive:

- internal (coming from the same region)
- external (coming from another region)

When a message is internal, the CIST part is received by the CIST, and each MST instance receives its respective M-record.

When a message is external, it is received only by the CIST. If the CIST role is root or alternate, or if the external BPDU is a topology change, it could have an impact on the MST instances.

An MST region includes both devices and LANs. A segment belongs to the region of its designated port. Therefore, a port in a different region than the designated port for a segment is a boundary port. This definition allows two ports internal to a region to share a segment with a port belonging to a different region, creating the possibility of a port receiving both internal and external messages.

The primary change from the Cisco prestandard implementation is that a designated port is not defined as boundary, unless it is running in an STP-compatible mode.



Note If there is a legacy STP device on the segment, messages are always considered external.

The other change from the Cisco prestandard implementation is that the CIST regional root device ID field is now inserted where an RSTP or legacy IEEE 802.1Q device has the sender device ID. The whole region performs like a single virtual device by sending a consistent sender device ID to neighboring devices. In this example, device C would receive a BPDU with the same consistent sender device ID of root, whether or not A or B is designated for the segment.

IEEE 802.1s Implementation

The Cisco implementation of the IEEE MST standard includes features required to meet the standard, as well as some of the desirable prestandard functionality that is not yet incorporated into the published standard.

Port Role Naming Change

The boundary role is no longer in the final MST standard, but this boundary concept is maintained in Cisco's implementation. However, an MST instance port at a boundary of the region might not follow the state of the corresponding CIST port. Two boundary roles currently exist:

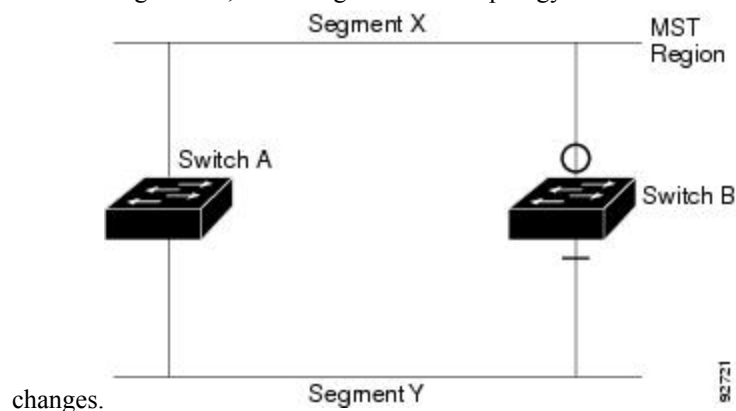
- The boundary port is the root port of the CIST regional root—When the CIST instance port is proposed and is in sync, it can send back an agreement and move to the forwarding state only after all the corresponding MSTI ports are in sync (and thus forwarding). The MSTI ports now have a special *primary* role.
- The boundary port is not the root port of the CIST regional root—The MSTI ports follow the state and role of the CIST port. The standard provides less information, and it might be difficult to understand why an MSTI port can be alternately blocking when it receives no BPDUs (MRecords). In this case, although the boundary role no longer exists, the **show** commands identify a port as boundary in the *type* column of the output.

Interoperation Between Legacy and Standard Devices

Because automatic detection of prestandard devices can fail, you can use an interface configuration command to identify prestandard ports. A region cannot be formed between a standard and a prestandard device, but they can interoperate by using the CIST. Only the capability of load-balancing over different instances is lost in that particular case. The CLI displays different flags depending on the port configuration when a port receives prestandard BPDUs. A syslog message also appears the first time a device receives a prestandard BPDU on a port that has not been configured for prestandard BPDU transmission.

Figure 26: Standard and Prestandard Device Interoperation

Assume that A is a standard device and B a prestandard device, both configured to be in the same region. A is the root device for the CIST, and B has a root port (BX) on segment X and an alternate port (BY) on segment Y. If segment Y flaps, and the port on BY becomes the alternate before sending out a single prestandard BPDU, AY cannot detect that a prestandard device is connected to Y and continues to send standard BPDUs. The port BY is fixed in a boundary, and no load balancing is possible between A and B. The same problem exists on segment X, but B might transmit topology



Note We recommend that you minimize the interaction between standard and prestandard MST implementations.

Detecting Unidirectional Link Failure

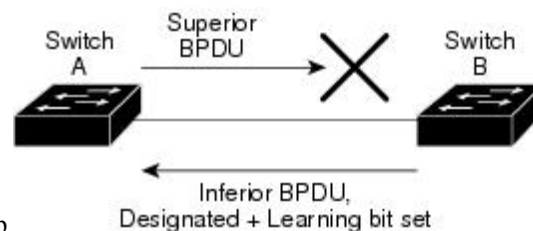
This feature is not yet present in the IEEE MST standard, but it is included in this Cisco IOS release. The software checks the consistency of the port role and state in the received BPDUs to detect unidirectional link failures that could cause bridging loops.

When a designated port detects a conflict, it keeps its role, but reverts to the discarding state because disrupting connectivity in case of inconsistency is preferable to opening a bridging loop.

Figure 27: Detecting Unidirectional Link Failure

This figure illustrates a unidirectional link failure that typically creates a bridging loop. Device A is the root device, and its BPDUs are lost on the link leading to device B. RSTP and MST BPDUs include the role and state of the sending port. With this information, device A can detect that device B does not react to the superior

BPDUs it sends and that device B is the designated, not root device. As a result, device A blocks (or keeps



blocking) its port, which prevents the bridging loop.

MSTP and Device Stacks

A device stack appears as a single spanning-tree node to the rest of the network, and all stack members use the same bridge ID for a given spanning tree. The bridge ID is derived from the MAC address of the active stack.

If a device that does not support MSTP is added to a device stack that does support MSTP or the reverse, the device is put into a version mismatch state. If possible, the device is automatically upgraded or downgraded to the same version of software that is running on the device stack.

When a new device joins the stack, it sets its device ID to the device ID. If the newly added device has the lowest ID and if the root path cost is the same among all stack members, the newly added device becomes the stack root. A topology change occurs if the newly added device contains a better root port for the device stack or a better designated port for the LAN connected to the stack. The newly added device causes a topology change in the network if another device connected to the newly added device changes its root port or designated ports.

When a stack member leaves the stack, spanning-tree reconvergence occurs within the stack (and possibly outside the stack). The remaining stack member with the lowest stack port ID becomes the stack root.

If the active stack fails or leaves the stack, the stack members elect a new active stack, and all stack members change the device IDs of the spanning trees to the new active device ID.

Interoperability with IEEE 802.1D STP

A device running MSTP supports a built-in protocol migration mechanism that enables it to interoperate with legacy IEEE 802.1D devices. If this device receives a legacy IEEE 802.1D configuration BPDU (a BPDU with the protocol version set to 0), it sends only IEEE 802.1D BPDUs on that port. An MSTP device also can detect that a port is at the boundary of a region when it receives a legacy BPDU, an MSTP BPDU (Version 3) associated with a different region, or an RSTP BPDU (Version 2).

However, the device does not automatically revert to the MSTP mode if it no longer receives IEEE 802.1D BPDUs because it cannot detect whether the legacy device has been removed from the link unless the legacy device is the designated device. A device might also continue to assign a boundary role to a port when the device to which this device is connected has joined the region. To restart the protocol migration process (force the renegotiation with neighboring devices), use the **clear spanning-tree detected-protocols** privileged EXEC command.

If all the legacy devices on the link are RSTP devices, they can process MSTP BPDUs as if they are RSTP BPDUs. Therefore, MSTP devices send either a Version 0 configuration and TCN BPDUs or Version 3 MSTP BPDUs on a boundary port. A boundary port connects to a LAN, the designated device of which is either a single spanning-tree device or a device with a different MST configuration.

RSTP Overview

The RSTP takes advantage of point-to-point wiring and provides rapid convergence of the spanning tree. Reconfiguration of the spanning tree can occur in less than 1 second (in contrast to 50 seconds with the default settings in the IEEE 802.1D spanning tree).

Port Roles and the Active Topology

The RSTP provides rapid convergence of the spanning tree by assigning port roles and by learning the active topology. The RSTP builds upon the IEEE 802.1D STP to select the device with the highest device priority (lowest numerical priority value) as the root device. The RSTP then assigns one of these port roles to individual ports:

- Root port—Provides the best path (lowest cost) when the device forwards packets to the root device.
- Designated port—Connects to the designated device, which incurs the lowest path cost when forwarding packets from that LAN to the root device. The port through which the designated device is attached to the LAN is called the designated port.
- Alternate port—Offers an alternate path toward the root device to that provided by the current root port.
- Backup port—Acts as a backup for the path provided by a designated port toward the leaves of the spanning tree. A backup port can exist only when two ports are connected in a loopback by a point-to-point link or when a device has two or more connections to a shared LAN segment.
- Disabled port—Has no role within the operation of the spanning tree.

A port with the root or a designated port role is included in the active topology. A port with the alternate or backup port role is excluded from the active topology.

In a stable topology with consistent port roles throughout the network, the RSTP ensures that every root port and designated port immediately transition to the forwarding state while all alternate and backup ports are always in the discarding state (equivalent to blocking in IEEE 802.1D). The port state controls the operation of the forwarding and learning processes.

Table 49: Port State Comparison

Operational Status	STP Port State (IEEE 802.1D)	RSTP Port State	Is Port Included in the Active Topology?
Enabled	Blocking	Discarding	No
Enabled	Listening	Discarding	No
Enabled	Learning	Learning	Yes
Enabled	Forwarding	Forwarding	Yes
Disabled	Disabled	Discarding	No

To be consistent with Cisco STP implementations, this guide defines the port state as *blocking* instead of *discarding*. Designated ports start in the listening state.

Rapid Convergence

The RSTP provides for rapid recovery of connectivity following the failure of a device, a device port, or a LAN. It provides rapid convergence for edge ports, new root ports, and ports connected through point-to-point links as follows:

- **Edge ports**—If you configure a port as an edge port on an RSTP device by using the **spanning-tree portfast** interface configuration command, the edge port immediately transitions to the forwarding state. An edge port is the same as a Port Fast-enabled port, and you should enable it only on ports that connect to a single end station.
- **Root ports**—If the RSTP selects a new root port, it blocks the old root port and immediately transitions the new root port to the forwarding state.
- **Point-to-point links**—If you connect a port to another port through a point-to-point link and the local port becomes a designated port, it negotiates a rapid transition with the other port by using the proposal-agreement handshake to ensure a loop-free topology.

Figure 28: Proposal and Agreement Handshaking for Rapid Convergence

Device A is connected to Device B through a point-to-point link, and all of the ports are in the blocking state. Assume that the priority of Device A is a smaller numerical value than the priority of Device B. Device A sends a proposal message (a configuration BPDU with the proposal flag set) to Device B, proposing itself as the designated device.

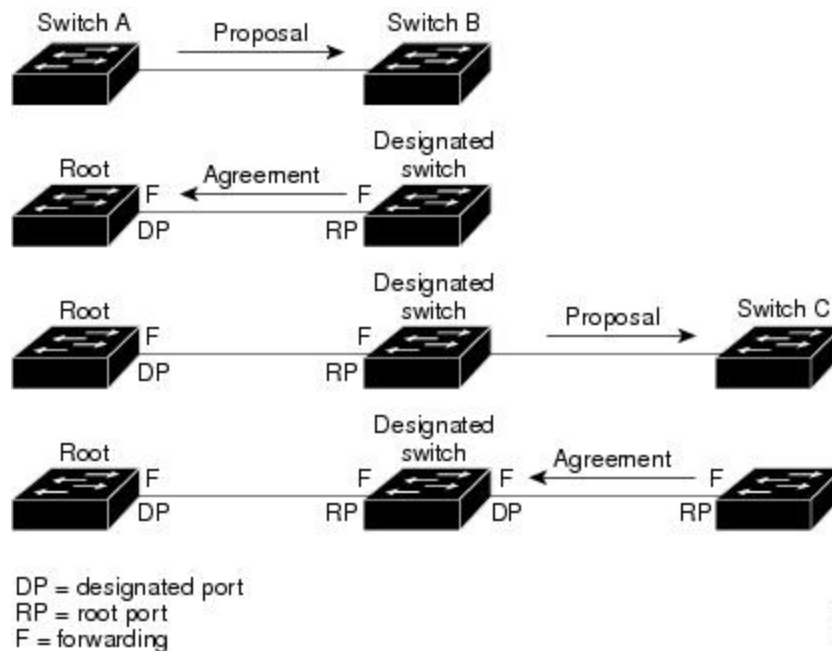
After receiving the proposal message, Device B selects as its new root port the port from which the proposal message was received, forces all nonedge ports to the blocking state, and sends an agreement message (a BPDU with the agreement flag set) through its new root port.

After receiving Device B's agreement message, Device A also immediately transitions its designated port to the forwarding state. No loops in the network are formed because Device B blocked all of its nonedge ports and because there is a point-to-point link between Devices A and B.

When Device C is connected to Device B, a similar set of handshaking messages are exchanged. Device C selects the port connected to Device B as its root port, and both ends immediately transition to the forwarding state. With each iteration of this handshaking process, one more device joins the active topology. As the network converges, this proposal-agreement handshaking progresses from the root toward the leaves of the spanning tree.

In a device stack, the cross-stack rapid transition (CSRT) feature ensures that a stack member receives acknowledgments from all stack members during the proposal-agreement handshaking before moving the port to the forwarding state. CSRT is automatically enabled when the device is in MST mode.

The device learns the link type from the port duplex mode: a full-duplex port is considered to have a point-to-point connection; a half-duplex port is considered to have a shared connection. You can override the default setting that is controlled by the duplex setting by using the **spanning-tree link-type** interface configuration command.



88790

Synchronization of Port Roles

When the device receives a proposal message on one of its ports and that port is selected as the new root port, the RSTP forces all other ports to synchronize with the new root information.

The device is synchronized with superior root information received on the root port if all other ports are synchronized. An individual port on the device is synchronized if

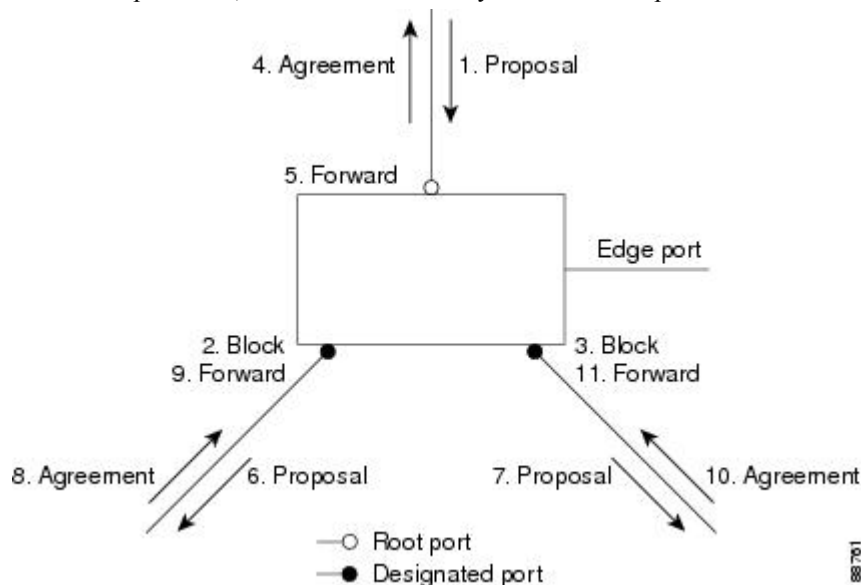
- That port is in the blocking state.
- It is an edge port (a port configured to be at the edge of the network).

If a designated port is in the forwarding state and is not configured as an edge port, it transitions to the blocking state when the RSTP forces it to synchronize with new root information. In general, when the RSTP forces a port to synchronize with root information and the port does not satisfy any of the above conditions, its port state is set to blocking.

Figure 29: Sequence of Events During Rapid Convergence

After ensuring that all of the ports are synchronized, the device sends an agreement message to the designated device corresponding to its root port. When the devices connected by a point-to-point link are in agreement

about their port roles, the RSTP immediately transitions the port states to forwarding.



Bridge Protocol Data Unit Format and Processing

The RSTP BPDU format is the same as the IEEE 802.1D BPDU format except that the protocol version is set to 2. A new 1-byte Version 1 Length field is set to zero, which means that no version 1 protocol information is present.

Table 50: RSTP BPDU Flags

Bit	Function
0	Topology change (TC)
1	Proposal
2–3:	Port role:
00	Unknown
01	Alternate port
10	Root port
11	Designated port
4	Learning
5	Forwarding
6	Agreement
7	Topology change acknowledgement (TCA)

The sending device sets the proposal flag in the RSTP BPDU to propose itself as the designated device on that LAN. The port role in the proposal message is always set to the designated port.

The sending device sets the agreement flag in the RSTP BPDU to accept the previous proposal. The port role in the agreement message is always set to the root port.

The RSTP does not have a separate topology change notification (TCN) BPDU. It uses the topology change (TC) flag to show the topology changes. However, for interoperability with IEEE 802.1D devices, the RSTP device processes and generates TCN BPDUs.

The learning and forwarding flags are set according to the state of the sending port.

Processing Superior BPDU Information

If a port receives superior root information (lower device ID, lower path cost, and so forth) than currently stored for the port, the RSTP triggers a reconfiguration. If the port is proposed and is selected as the new root port, RSTP forces all the other ports to synchronize.

If the BPDU received is an RSTP BPDU with the proposal flag set, the device sends an agreement message after all of the other ports are synchronized. If the BPDU is an IEEE 802.1D BPDU, the device does not set the proposal flag and starts the forward-delay timer for the port. The new root port requires twice the forward-delay time to transition to the forwarding state.

If the superior information received on the port causes the port to become a backup or alternate port, RSTP sets the port to the blocking state but does not send the agreement message. The designated port continues sending BPDUs with the proposal flag set until the forward-delay timer expires, at which time the port transitions to the forwarding state.

Processing Inferior BPDU Information

If a designated port receives an inferior BPDU (such as a higher device ID or a higher path cost than currently stored for the port) with a designated port role, it immediately replies with its own information.

Topology Changes

This section describes the differences between the RSTP and the IEEE 802.1D in handling spanning-tree topology changes.

- **Detection**—Unlike IEEE 802.1D in which *any* transition between the blocking and the forwarding state causes a topology change, *only* transitions from the blocking to the forwarding state cause a topology change with RSTP (only an increase in connectivity is considered a topology change). State changes on an edge port do not cause a topology change. When an RSTP device detects a topology change, it deletes the learned information on all of its nonedge ports except on those from which it received the TC notification.
- **Notification**—Unlike IEEE 802.1D, which uses TCN BPDUs, the RSTP does not use them. However, for IEEE 802.1D interoperability, an RSTP device processes and generates TCN BPDUs.
- **Acknowledgement**—When an RSTP device receives a TCN message on a designated port from an IEEE 802.1D device, it replies with an IEEE 802.1D configuration BPDU with the TCA bit set. However, if the TC-while timer (the same as the topology-change timer in IEEE 802.1D) is active on a root port connected to an IEEE 802.1D device and a configuration BPDU with the TCA bit set is received, the TC-while timer is reset.

This behavior is only required to support IEEE 802.1D devices. The RSTP BPDUs never have the TCA bit set.

- **Propagation**—When an RSTP device receives a TC message from another device through a designated or root port, it propagates the change to all of its nonedge, designated ports and to the root port (excluding

the port on which it is received). The device starts the TC-while timer for all such ports and flushes the information learned on them.

- Protocol migration—For backward compatibility with IEEE 802.1D devices, RSTP selectively sends IEEE 802.1D configuration BPDUs and TCN BPDUs on a per-port basis.

When a port is initialized, the migrate-delay timer is started (specifies the minimum time during which RSTP BPDUs are sent), and RSTP BPDUs are sent. While this timer is active, the device processes all BPDUs received on that port and ignores the protocol type.

If the device receives an IEEE 802.1D BPDU after the port migration-delay timer has expired, it assumes that it is connected to an IEEE 802.1D device and starts using only IEEE 802.1D BPDUs. However, if the RSTP device is using IEEE 802.1D BPDUs on a port and receives an RSTP BPDU after the timer has expired, it restarts the timer and starts using RSTP BPDUs on that port.

Protocol Migration Process

A device running MSTP supports a built-in protocol migration mechanism that enables it to interoperate with legacy IEEE 802.1D devices. If this device receives a legacy IEEE 802.1D configuration BPDU (a BPDU with the protocol version set to 0), it sends only IEEE 802.1D BPDUs on that port. An MSTP device also can detect that a port is at the boundary of a region when it receives a legacy BPDU, an MST BPDU (Version 3) associated with a different region, or an RST BPDU (Version 2).

However, the device does not automatically revert to the MSTP mode if it no longer receives IEEE 802.1D BPDUs because it cannot detect whether the legacy device has been removed from the link unless the legacy device is the designated device. A device also might continue to assign a boundary role to a port when the device to which it is connected has joined the region.

Default MSTP Configuration

Table 51: Default MSTP Configuration

Feature	Default Setting
Spanning-tree mode	MSTP
Device priority (configurable on a per-CIST port basis)	32768
Spanning-tree port priority (configurable on a per-CIST port basis)	128
Spanning-tree port cost (configurable on a per-CIST port basis)	1000 Mb/s: 20000 100 Mb/s: 20000 10 Mb/s: 20000 1000 Mb/s: 20000 100 Mb/s: 20000 10 Mb/s: 20000
Hello time	3 seconds

Feature	Default Setting
Forward-delay time	20 seconds
Maximum-aging time	20 seconds
Maximum hop count	20 hops

About MST-to-PVST+ Interoperability (PVST+ Simulation)

The PVST+ simulation feature enables seamless interoperability between MST and Rapid PVST+. You can enable or disable this per port, or globally. PVST+ simulation is enabled by default.

However, you may want to control the connection between MST and Rapid PVST+ to protect against accidentally connecting an MST-enabled port to a Rapid PVST+-enabled port. Because Rapid PVST+ is the default STP mode, you may encounter many Rapid PVST+-enabled connections.

Disabling this feature causes the switch to stop the MST region from interacting with PVST+ regions. The MST-enabled port moves to a PVST peer inconsistent (blocking) state once it detects it is connected to a Rapid PVST+-enabled port. This port remains in the inconsistent state until the port stops receiving Shared Spanning Tree Protocol (SSTP) BPDUs, and then the port resumes the normal STP transition process.

You can for instance, disable PVST+ simulation, to prevent an incorrectly configured switch from connecting to a network where the STP mode is not MSTP (the default mode is PVST+).

Observe these guidelines when you configure MST switches (in the same region) to interact with PVST+ switches:

- Configure the root for all VLANs inside the MST region as shown in this example:

```
Switch# show spanning-tree mst interface gigabitEthernet 1/1
GigabitEthernet1/1 of MST00 is root forwarding
Edge port: no (trunk) port guard : none (default)
Link type: point-to-point (auto) bpdu filter: disable (default)
Boundary : boundary (PVST) bpdu guard : disable (default)
Bpdus sent 10, received 310
```

```
Instance Role Sts Cost Prio.Nbr Vlans mapped
-----
0 Root FWD 20000 128.1 1-2,4-2999,4000-4094
3 Boun FWD 20000 128.1 3,3000-3999
```

The ports that belong to the MST switch at the boundary simulate PVST+ and send PVST+ BPDUs for all the VLANs.

If you enable loop guard on the PVST+ switches, the ports might change to a loop-inconsistent state when the MST switches change their configuration. To correct the loop-inconsistent state, you must disable and re-enable loop guard on that PVST+ switch.

- Do not locate the root for some or all of the VLANs inside the PVST+ side of the MST switch because when the MST switch at the boundary receives PVST+ BPDUs for all or some of the VLANs on its designated ports, root guard sets the port to the blocking state.
- When you connect a PVST+ switch to two different MST regions, the topology change from the PVST+ switch does not pass beyond the first MST region. In such a case, the topology changes are propagated only in the instance to which the VLAN is mapped. The topology change stays local to the first MST region, and the Cisco Access Manager (CAM) entries in the other region are not flushed. To make the

topology change visible throughout other MST regions, you can map that VLAN to IST or connect the PVST+ switch to the two regions through access links.

- When you disable the PVST+ simulation, note that the PVST+ peer inconsistency can also occur while the port is already in other states of inconsistency. For example, the root bridge for all STP instances must all be in either the MST region or the Rapid PVST+ side. If the root bridge for all STP instances are not on one side or the other, the software moves the port into a PVST + simulation-inconsistent state.



Note We recommend that you put the root bridge for all STP instances in the MST region.

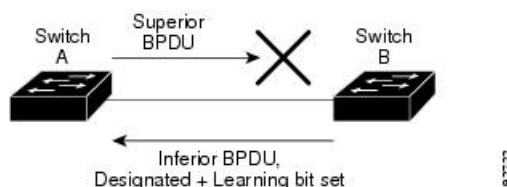
About Detecting Unidirectional Link Failure

The dispute mechanism that detects unidirectional link failures is included in the IEEE 802.1D-2004 RSTP and IEEE 802.1Q-2005 MSTP standard, and requires no user configuration.

The switch checks the consistency of the port role and state in the BPDUs it receives, to detect unidirectional link failures that could cause bridging loops. When a designated port detects a conflict, it keeps its role, but reverts to a discarding (blocking) state because disrupting connectivity in case of inconsistency is preferable to opening a bridging loop.

For example, in the figure below, Switch A is the root bridge and Switch B is the designated port. BPDUs from Switch A are lost on the link leading to switch B.

Figure 30: Detecting Unidirectional Link Failure

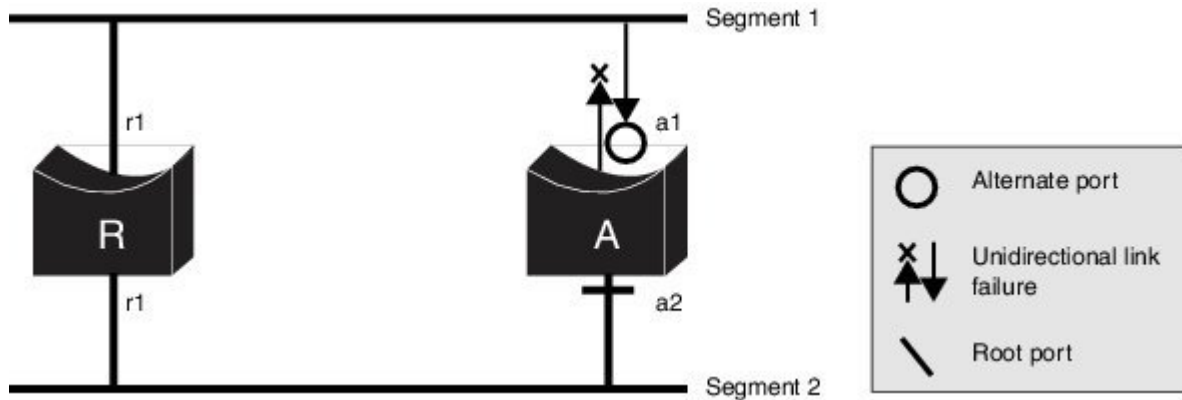


Since Rapid PVST+ (802.1w) and MST BPDUs include the role and state of the sending port, Switch A detects (from the inferior BPDU), that switch B does not react to the superior BPDUs it sends, because switch B has the role of a designated port and not the root bridge. As a result, switch A blocks (or keeps blocking) its port, thus preventing the bridging loop.

Note these guidelines and limitations relating to the dispute mechanism:

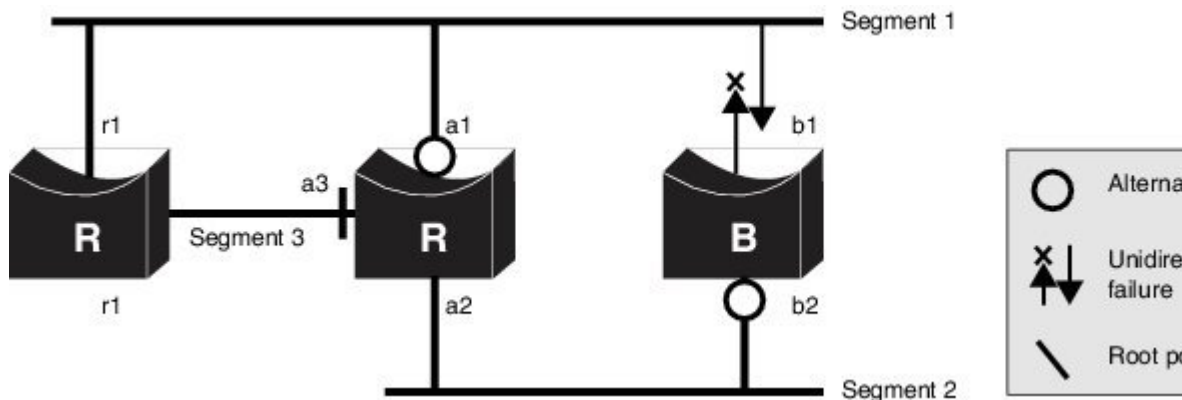
- It works only on switches running RSTP or MST (the dispute mechanism requires reading the role and state of the port initiating BPDUs).
- It may result in loss of connectivity. For example, in the figure below, Bridge A cannot transmit on the port it elected as a root port. As a result of this situation, there is loss of connectivity (r1 and r2 are designated, a1 is root and a2 is alternate. There is only a one way connectivity between A and R).

Figure 31: Loss of Connectivity



- It may cause permanent bridging loops on shared segments. For example, in the figure below, suppose that bridge R has the best priority, and that port b1 cannot receive any traffic from the shared segment 1 and sends inferior designated information on segment 1. Both r1 and a1 can detect this inconsistency. However, with the current dispute mechanism, only r1 will revert to discarding while the root port a1 opens a permanent loop. However, this problem does not occur in Layer 2 switched networks that are connected by point-to-point links.

Figure 32: Bridging Loops on Shared Segments



How to Configure MSTP Features

Specifying the MST Region Configuration and Enabling MSTP

For two or more switches to be in the same MST region, they must have the same VLAN-to-instance mapping, the same configuration revision number, and the same name.

A region can have one member or multiple members with the same MST configuration; each member must be capable of processing RSTP BPDUs. There is no limit to the number of MST regions in a network, but each region can only support up to 65 spanning-tree instances. You can assign a VLAN to only one spanning-tree instance at a time.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **spanning-tree mst configuration**
4. **instance** *instance-id* **vlan** *vlan-range*
5. **name** *name*
6. **revision** *version*
7. **show pending**
8. **exit**
9. **spanning-tree mode mst**
10. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	spanning-tree mst configuration Example: Device(config)# spanning-tree mst configuration	Enters MST configuration mode.
Step 4	instance <i>instance-id</i> vlan <i>vlan-range</i> Example: Device(config-mst)# instance 1 vlan 10-20	Maps VLANs to an MST instance. <ul style="list-style-type: none"> • For <i>instance-id</i>, the range is 0 to 4094. • For vlan <i>vlan-range</i>, the range is 1 to 4094. When you map VLANs to an MST instance, the mapping is incremental, and the VLANs specified in the command are added to or removed from the VLANs that were previously mapped. <p>To specify a VLAN range, use a hyphen; for example, instance 1 vlan 1-63 maps VLANs 1 through 63 to MST instance 1.</p> <p>To specify a VLAN series, use a comma; for example, instance 1 vlan 10, 20, 30 maps VLANs 10, 20, and 30 to MST instance 1.</p>

	Command or Action	Purpose
Step 5	name <i>name</i> Example: Device(config-mst)# name region1	Specifies the configuration name. The <i>name</i> string has a maximum length of 32 characters and is case sensitive.
Step 6	revision <i>version</i> Example: Device(config-mst)# revision 1	Specifies the configuration revision number. The range is 0 to 65535.
Step 7	show pending Example: Device(config-mst)# show pending	Verifies your configuration by displaying the pending configuration.
Step 8	exit Example: Device(config-mst)# exit	Applies all changes, and returns to global configuration mode.
Step 9	spanning-tree mode mst Example: Device(config)# spanning-tree mode mst	Enables MSTP. RSTP is also enabled. Changing spanning-tree modes can disrupt traffic because all spanning-tree instances are stopped for the previous mode and restarted in the new mode. You cannot run both MSTP and PVST+ or both MSTP and Rapid PVST+ at the same time.
Step 10	end Example: Device(config)# end	Returns to privileged EXEC mode.

Configuring the Root Device

This procedure is optional.

Before you begin

A multiple spanning tree (MST) must be specified and enabled on the device. For instructions, see Related Topics.

You must also know the specified MST instance ID. Step 2 in the example uses 0 as the instance ID because that was the instance ID set up by the instructions listed under Related Topics.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **spanning-tree mst *instance-id* root primary**
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	spanning-tree mst <i>instance-id</i> root primary Example: Device(config)# spanning-tree mst 0 root primary	Configures a device as the root device. <ul style="list-style-type: none"> • For <i>instance-id</i>, you can specify a single instance, a range of instances separated by a hyphen, or a series of instances separated by a comma. The range is 0 to 4094.
Step 4	end Example: Device(config)# end	Returns to privileged EXEC mode.

Configuring a Secondary Root Device

When you configure a device with the extended system ID support as the secondary root, the device priority is modified from the default value (32768) to 28672. The device is then likely to become the root device for the specified instance if the primary root device fails. This is assuming that the other network devices use the default device priority of 32768 and therefore are unlikely to become the root device.

You can execute this command on more than one device to configure multiple backup root devices. Use the same network diameter and hello-time values that you used when you configured the primary root device with the **spanning-tree mst *instance-id* root primary** global configuration command.

This procedure is optional.

Before you begin

A multiple spanning tree (MST) must be specified and enabled on the device. For instructions, see Related Topics.

You must also know the specified MST instance ID. This example uses 0 as the instance ID because that was the instance ID set up by the instructions listed under Related Topics.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **spanning-tree mst *instance-id* root secondary**
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	spanning-tree mst <i>instance-id</i> root secondary Example: Device(config)# spanning-tree mst 0 root secondary	Configures a device as the secondary root device. <ul style="list-style-type: none"> • For <i>instance-id</i>, you can specify a single instance, a range of instances separated by a hyphen, or a series of instances separated by a comma. The range is 0 to 4094.
Step 4	end Example: Device(config)# end	Returns to privileged EXEC mode.

Configuring Port Priority

If a loop occurs, the MSTP uses the port priority when selecting an interface to put into the forwarding state. You can assign higher priority values (lower numerical values) to interfaces that you want selected first and lower priority values (higher numerical values) that you want selected last. If all interfaces have the same priority value, the MSTP puts the interface with the lowest interface number in the forwarding state and blocks the other interfaces.



Note If the device is a member of a device stack, you must use the **spanning-tree mst [instance-id] cost cost** interface configuration command instead of the **spanning-tree mst [instance-id] port-priority priority** interface configuration command to select a port to put in the forwarding state. Assign lower cost values to ports that you want selected first and higher cost values to ports that you want selected last. For more information, see the path costs topic listed under Related Topics.

This procedure is optional.

Before you begin

A multiple spanning tree (MST) must be specified and enabled on the device. For instructions, see Related Topics.

You must also know the specified MST instance ID and the interface used. This example uses 0 as the instance ID and GigabitEthernet0/1 as the interface because that was the instance ID and interface set up by the instructions listed under Related Topics.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **spanning-tree mst** *instance-id* **port-priority** *priority*
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	interface <i>interface-id</i> Example: Device(config)# interface gigabitethernet 1/0/1	Specifies an interface to configure, and enters interface configuration mode.
Step 4	spanning-tree mst <i>instance-id</i> port-priority <i>priority</i> Example:	Configures port priority. <ul style="list-style-type: none"> • For <i>instance-id</i>, you can specify a single instance, a range of instances separated by a hyphen, or a series

	Command or Action	Purpose
	<pre>Device(config-if) # spanning-tree mst 0 port-priority 64</pre>	<p>of instances separated by a comma. The range is 0 to 4094.</p> <ul style="list-style-type: none"> For <i>priority</i>, the range is 0 to 240 in increments of 16. The default is 128. The lower the number, the higher the priority. <p>The priority values are 0, 16, 32, 48, 64, 80, 96, 112, 128, 144, 160, 176, 192, 208, 224, and 240. All other values are rejected.</p>
Step 5	<p>end</p> <p>Example:</p> <pre>Device(config-if) # end</pre>	Returns to privileged EXEC mode.

The **show spanning-tree mst interface** *interface-id* privileged EXEC command displays information only if the port is in a link-up operative state. Otherwise, you can use the **show running-config interface** privileged EXEC command to confirm the configuration.

Configuring Path Cost

The MSTP path cost default value is derived from the media speed of an interface. If a loop occurs, the MSTP uses cost when selecting an interface to put in the forwarding state. You can assign lower cost values to interfaces that you want selected first and higher cost values that you want selected last. If all interfaces have the same cost value, the MSTP puts the interface with the lowest interface number in the forwarding state and blocks the other interfaces.

This procedure is optional.

Before you begin

A multiple spanning tree (MST) must be specified and enabled on the device. For instructions, see Related Topics.

You must also know the specified MST instance ID and the interface used. This example uses 0 as the instance ID and GigabitEthernet1/0/1 as the interface because that was the instance ID and interface set up by the instructions listed under Related Topics.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **spanning-tree mst** *instance-id* **cost** *cost*
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	interface <i>interface-id</i> Example: Device(config)# interface gigabitethernet 1/0/1	Specifies an interface to configure, and enters interface configuration mode. Valid interfaces include physical ports and port-channel logical interfaces. The port-channel range is 1 to 48.
Step 4	spanning-tree mst <i>instance-id</i> cost <i>cost</i> Example: Device(config-if)# spanning-tree mst 0 cost 17031970	Configures the cost. If a loop occurs, the MSTP uses the path cost when selecting an interface to place into the forwarding state. A lower path cost represents higher-speed transmission. <ul style="list-style-type: none"> • For <i>instance-id</i>, you can specify a single instance, a range of instances separated by a hyphen, or a series of instances separated by a comma. The range is 0 to 4094. • For <i>cost</i>, the range is 1 to 200000000; the default value is derived from the media speed of the interface.
Step 5	end Example: Device(config-if)# end	Returns to privileged EXEC mode.

The **show spanning-tree mst interface *interface-id*** privileged EXEC command displays information only for ports that are in a link-up operative state. Otherwise, you can use the **show running-config** privileged EXEC command to confirm the configuration.

Configuring the Device Priority

Changing the priority of a device makes it more likely to be chosen as the root device whether it is a standalone device or a device in the stack.



Note Exercise care when using this command. For normal network configurations, we recommend that you use the **spanning-tree mst *instance-id* root primary** and the **spanning-tree mst *instance-id* root secondary** global configuration commands to specify a device as the root or secondary root device. You should modify the device priority only in circumstances where these commands do not work.

This procedure is optional.

Before you begin

A multiple spanning tree (MST) must be specified and enabled on the device. For instructions, see Related Topics.

You must also know the specified MST instance ID used. This example uses 0 as the instance ID because that was the instance ID set up by the instructions listed under Related Topics.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **spanning-tree mst *instance-id* priority *priority***
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	spanning-tree mst <i>instance-id</i> priority <i>priority</i> Example: Device(config)# spanning-tree mst 0 priority 40960	Configures the device priority. <ul style="list-style-type: none"> • For <i>instance-id</i>, you can specify a single instance, a range of instances separated by a hyphen, or a series of instances separated by a comma. The range is 0 to 4094. • For <i>priority</i>, the range is 0 to 61440 in increments of 4096; the default is 32768. The lower the number, the more likely the device will be chosen as the root device.

	Command or Action	Purpose
		Priority values are 0, 4096, 8192, 12288, 16384, 20480, 24576, 28672, 32768, 36864, 40960, 45056, 49152, 53248, 57344, and 61440. These are the only acceptable values.
Step 4	end Example: Device(config-if)# end	Returns to privileged EXEC mode.

Configuring the Hello Time

The hello time is the time interval between configuration messages generated and sent by the root device. This procedure is optional.

Before you begin

A multiple spanning tree (MST) must be specified and enabled on the device. For instructions, see Related Topics.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **spanning-tree mst hello-time *seconds***
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	spanning-tree mst hello-time <i>seconds</i> Example: Device(config)# spanning-tree mst hello-time 4	Configures the hello time for all MST instances. The hello time is the time interval between configuration messages generated and sent by the root device. These messages indicate that the device is alive.

	Command or Action	Purpose
		For <i>seconds</i> , the range is 1 to 10; the default is 3.
Step 4	end Example: Device(config)# end	Returns to privileged EXEC mode.

Configuring the Forwarding-Delay Time

Before you begin

A multiple spanning tree (MST) must be specified and enabled on the device. For instructions, see Related Topics.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **spanning-tree mst forward-time *seconds***
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	spanning-tree mst forward-time <i>seconds</i> Example: Device(config)# spanning-tree mst forward-time 25	Configures the forward time for all MST instances. The forwarding delay is the number of seconds a port waits before changing from its spanning-tree learning and listening states to the forwarding state. For <i>seconds</i> , the range is 4 to 30; the default is 20.
Step 4	end Example:	Returns to privileged EXEC mode.

	Command or Action	Purpose
	Device(config)# end	

Configuring the Maximum-Aging Time

Before you begin

A multiple spanning tree (MST) must be specified and enabled on the device. For instructions, see Related Topics.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **spanning-tree mst max-age *seconds***
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none">• Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	spanning-tree mst max-age <i>seconds</i> Example: Device(config)# spanning-tree mst max-age 40	Configures the maximum-aging time for all MST instances. The maximum-aging time is the number of seconds a device waits without receiving spanning-tree configuration messages before attempting a reconfiguration. For <i>seconds</i> , the range is 6 to 40; the default is 20.
Step 4	end Example: Device(config)# end	Returns to privileged EXEC mode.

Configuring the Maximum-Hop Count

This procedure is optional.

Before you begin

A multiple spanning tree (MST) must be specified and enabled on the device. For instructions, see Related Topics.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **spanning-tree mst max-hops *hop-count***
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	spanning-tree mst max-hops <i>hop-count</i> Example: Device(config)# spanning-tree mst max-hops 25	Specifies the number of hops in a region before the BPDU is discarded, and the information held for a port is aged. For <i>hop-count</i> , the range is 1 to 255; the default is 20.
Step 4	end Example: Device(config)# end	Returns to privileged EXEC mode.

Specifying the Link Type to Ensure Rapid Transitions

If you connect a port to another port through a point-to-point link and the local port becomes a designated port, the RSTP negotiates a rapid transition with the other port by using the proposal-agreement handshake to ensure a loop-free topology.

By default, the link type is controlled from the duplex mode of the interface: a full-duplex port is considered to have a point-to-point connection; a half-duplex port is considered to have a shared connection. If you have a half-duplex link physically connected point-to-point to a single port on a remote device running MSTP, you can override the default setting of the link type and enable rapid transitions to the forwarding state.

This procedure is optional.

Before you begin

A multiple spanning tree (MST) must be specified and enabled on the device. For instructions, see Related Topics.

You must also know the specified MST instance ID and the interface used. This example uses 0 as the instance ID and GigabitEthernet1/0/1 as the interface because that was the instance ID and interface set up by the instructions listed under Related Topics.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **spanning-tree link-type point-to-point**
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	interface <i>interface-id</i> Example: Device(config)# interface gigabitethernet 1/0/1	Specifies an interface to configure, and enters interface configuration mode. Valid interfaces include physical ports, VLANs, and port-channel logical interfaces. The VLAN ID range is 1 to 4094. The port-channel range is 1 to 48.
Step 4	spanning-tree link-type point-to-point Example: Device(config-if)# spanning-tree link-type point-to-point	Specifies that the link type of a port is point-to-point.

	Command or Action	Purpose
Step 5	end Example: Device(config-if)# end	Returns to privileged EXEC mode.

Designating the Neighbor Type

A topology could contain both prestandard and IEEE 802.1s standard compliant devices. By default, ports can automatically detect prestandard devices, but they can still receive both standard and prestandard BPDUs. When there is a mismatch between a device and its neighbor, only the CIST runs on the interface.

You can choose to set a port to send only prestandard BPDUs. The prestandard flag appears in all the **show** commands, even if the port is in STP compatibility mode.

This procedure is optional.

Before you begin

A multiple spanning tree (MST) must be specified and enabled on the device. For instructions, see Related Topics.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **spanning-tree mst pre-standard**
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	interface <i>interface-id</i> Example: Device(config)# interface gigabitethernet 1/0/1	Specifies an interface to configure, and enters interface configuration mode. Valid interfaces include physical ports.

	Command or Action	Purpose
Step 4	spanning-tree mst pre-standard Example: Device(config-if)# spanning-tree mst pre-standard	Specifies that the port can send only prestandard BPDUs.
Step 5	end Example: Device(config-if)# end	Returns to privileged EXEC mode.

Restarting the Protocol Migration Process

This procedure restarts the protocol migration process and forces renegotiation with neighboring devices. It reverts the device to MST mode. It is needed when the device no longer receives IEEE 802.1D BPDUs after it has been receiving them.

Follow these steps to restart the protocol migration process (force the renegotiation with neighboring devices) on the device.

Before you begin

A multiple spanning tree (MST) must be specified and enabled on the device. For instructions, see Related Topics.

If you want to use the interface version of the command, you must also know the MST interface used. This example uses GigabitEthernet1/0/1 as the interface because that was the interface set up by the instructions listed under Related Topics.

SUMMARY STEPS

1. **enable**
2. Enter one of the following commands:
 - **clear spanning-tree detected-protocols**
 - **clear spanning-tree detected-protocols interface *interface-id***

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	Enter one of the following commands: <ul style="list-style-type: none"> • clear spanning-tree detected-protocols 	The device reverts to the MSTP mode, and the protocol migration process restarts.

	Command or Action	Purpose
	<ul style="list-style-type: none"> • <code>clear spanning-tree detected-protocols interface interface-id</code> <p>Example:</p> <pre>Device# clear spanning-tree detected-protocols OR Device# clear spanning-tree detected-protocols interface gigabitethernet 1/0/1</pre>	

What to do next

This procedure may need to be repeated if the device receives more legacy IEEE 802.1D configuration BPDUs (BPDUs with the protocol version set to 0).

Configuring PVST+ Simulation

PVST+ simulation is enabled by default. This means that all ports automatically interoperate with a connected device that is running in Rapid PVST+ mode. If you disabled the feature and want to re-configure it, refer to the following tasks.

To enable PVST+ simulation globally, perform this task:

SUMMARY STEPS

1. `enable`
2. `configure terminal`
3. `spanning-tree mst simulate pvst global`
4. `end`

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p><code>enable</code></p> <p>Example:</p> <pre>Device> enable</pre>	<p>Enables privileged EXEC mode.</p> <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	<p><code>configure terminal</code></p> <p>Example:</p> <pre>Device# configure terminal</pre>	<p>Enters global configuration mode.</p>
Step 3	<p><code>spanning-tree mst simulate pvst global</code></p> <p>Example:</p>	<p>Enables PVST+ simulation globally.</p> <p>To prevent the switch from automatically interoperating with a connecting switch that is running Rapid PVST+, enter the no version of the command.</p>

	Command or Action	Purpose
	Device(config)# spanning-tree mst simulate pvst global	
Step 4	end Example: Device(config)# end	Returns to privileged EXEC mode.

Enabling PVST+ Simulation on a Port

To enable PVST+ simulation on a port, perform this task:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **spanning-tree mst simulate pvst**
5. **end**
6. **show spanning-tree summary**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	interface <i>interface-id</i> Example: Device(config)# interface gi1/0/1	Selects a port to configure.
Step 4	spanning-tree mst simulate pvst Example:	Enables PVST+ simulation on the specified interface. To prevent a specified interface from automatically interoperating with a connecting switch that is not running

	Command or Action	Purpose
	Device(config-if)# spanning-tree mst simulate pvst	MST, enter the spanning-tree mst simulate pvst disable command.
Step 5	end Example: Device(config)# end	Returns to privileged EXEC mode.
Step 6	show spanning-tree summary Example: Device# show spanning-tree summary	Verifies the configuration.

Examples

Examples: PVST+ Simulation

This example shows how to prevent the switch from automatically interoperating with a connecting switch that is running Rapid PVST+:

```
Switch# configure terminal
Switch(config)# no spanning-tree mst simulate pvst global
```

This example shows how to prevent a port from automatically interoperating with a connecting device that is running Rapid PVST+:

```
Switch(config)# interface1/0/1
Switch(config-if)# spanning-tree mst simulate pvst disable
```

The following sample output shows the system message you receive when a SSTP BPDU is received on a port and PVST+ simulation is disabled:

```
Message
SPANTREE_PVST_PEER_BLOCK: PVST BPDU detected on port %s [port number].
```

```
Severity
Critical
```

```
Explanation
A PVST+ peer was detected on the specified interface on the switch. PVST+
simulation feature is disabled, as a result of which the interface was
moved to the spanning tree
Blocking state.
```

```
Action
```

Identify the PVST+ switch from the network which might be configured incorrectly.

The following sample output shows the system message you receive when peer inconsistency on the interface is cleared:

```
Message
SPANTREE_PVST_PEER_UNBLOCK: Unblocking port %s [port number].
```

```
Severity
Critical
```

```
Explanation
The interface specified in the error message has been restored to normal
spanning tree state.
```

```
Action
None.
```

This example shows the spanning tree status when port **1/0/1** has been configured to disable PVST+ simulation and is currently in the peer type inconsistent state:

```
Switch# show spanning-tree
VLAN0010
  Spanning tree enabled protocol mstp
  Root ID Priority 32778
        Address 0002.172c.f400
        This bridge is the root
        Hello Time 2 sec Max Age 20 sec Forward Delay 15 sec
  Bridge ID Priority 32778 (priority 32768 sys-id-ext 10)
        Address 0002.172c.f400
        Hello Time 2 sec Max Age 20 sec Forward Delay 15 sec
        Aging Time 300
Interface      Role Sts Cost          Prio.Nbr Type
-----
Gi1/0/1       Desg BKN*4      128.270 P2p *PVST_Peer_Inc
```

This example shows the spanning tree summary when PVST+ simulation is enabled in the MSTP mode:

```
Switch# show spanning-tree summary
Switch is in mst mode (IEEE Standard)
Root bridge for: MST0
EtherChannel misconfig guard is enabled
Extended system ID is enabled
Portfast Default is disabled
PortFast BPDU Guard Default is disabled
Portfast BPDU Filter Default is disabled
Loopguard Default is disabled
UplinkFast is disabled
BackboneFast is disabled
Pathcost method used is long
PVST Simulation Default is enabled
Name          Blocking Listening Learning Forwarding STP Active
-----
MST0          2           0           0           0           2
```

```
-----
1 mst                2                0                0                0                2
```

This example shows the spanning tree summary when PVST+ simulation is disabled in any STP mode:

```
Switch# show spanning-tree summary
Switch is in mst mode (IEEE Standard)
Root bridge for: MST0
EtherChannel misconfig guard is enabled
Extended system ID is enabled
Portfast Default is disabled
PortFast BPDU Guard Default is disabled
Portfast BPDU Filter Default is disabled
Loopguard Default is disabled
UplinkFast is disabled
BackboneFast is disabled
Pathcost method used is long
PVST Simulation Default is disabled
```

Name	Blocking	Listening	Learning	Forwarding	STP Active
MST0	2	0	0	0	2
1 mst	2	0	0	0	2

This example shows the spanning tree summary when the switch is not in MSTP mode, that is, the switch is in PVST or Rapid-PVST mode. The output string displays the current STP mode:

```
Switch# show spanning-tree summary
Switch is in rapid-pvst mode
Root bridge for: VLAN0001, VLAN2001-VLAN2002
EtherChannel misconfig guard is enabled
Extended system ID is enabled
Portfast Default is disabled
PortFast BPDU Guard Default is disabled
Portfast BPDU Filter Default is disabled
Loopguard Default is disabled
UplinkFast is disabled
BackboneFast is disabled
Pathcost method used is short
PVST Simulation Default is enabled but inactive in rapid-pvst mode
```

Name	Blocking	Listening	Learning	Forwarding	STP Active
VLAN0001	2	0	0	0	2
VLAN2001	2	0	0	0	2
VLAN2002	2	0	0	0	2
3 vlans	6	0	0	0	6

This example shows the interface details when PVST+ simulation is globally enabled, or the default configuration:

```
Switch# show spanning-tree interface1/0/1 detail
```

```
Port 269 (GigabitEthernet1/0/1) of VLAN0002 is forwarding
  Port path cost 4, Port priority 128, Port Identifier 128.297.
  Designated root has priority 32769, address 0013.5f20.01c0
  Designated bridge has priority 32769, address 0013.5f20.01c0
  Designated port id is 128.297, designated path cost 0
  Timers: message age 0, forward delay 0, hold 0
  Number of transitions to forwarding state: 1
  Link type is point-to-point by default
  PVST Simulation is enabled by default
  BPDU: sent 132, received 1
```

This example shows the interface details when PVST+ simulation is globally disabled:

```
Switch# show spanning-tree interface1/0/1 detail
Port 269 (GigabitEthernet1/0/1) of VLAN0002 is forwarding
  Port path cost 4, Port priority 128, Port Identifier 128.297.
  Designated root has priority 32769, address 0013.5f20.01c0
  Designated bridge has priority 32769, address 0013.5f20.01c0
  Designated port id is 128.297, designated path cost 0
  Timers: message age 0, forward delay 0, hold 0
  Number of transitions to forwarding state: 1
  Link type is point-to-point by default
  PVST Simulation is disabled by default
  BPDU: sent 132, received 1
```

This example shows the interface details when PVST+ simulation is explicitly enabled on the port:

```
Switch# show spanning-tree interface1/0/1 detail
Port 269 (GigabitEthernet1/0/1) of VLAN0002 is forwarding
  Port path cost 4, Port priority 128, Port Identifier 128.297.
  Designated root has priority 32769, address 0013.5f20.01c0
  Designated bridge has priority 32769, address 0013.5f20.01c0
  Designated port id is 128.297, designated path cost 0
  Timers: message age 0, forward delay 0, hold 0
  Number of transitions to forwarding state: 1
  Link type is point-to-point by default
  PVST Simulation is enabled
  BPDU: sent 132, received 1
```

This example shows the interface details when the PVST+ simulation feature is disabled and a PVST Peer inconsistency has been detected on the port:

```
Switch# show spanning-tree interface1/0/1 detail
Port 269 (GigabitEthernet1/0/1) of VLAN0002 is broken (PVST Peer Inconsistent)
  Port path cost 4, Port priority 128, Port Identifier 128.297.
  Designated root has priority 32769, address 0013.5f20.01c0
  Designated bridge has priority 32769, address 0013.5f20.01c0
  Designated port id is 128.297, designated path cost 0
  Timers: message age 0, forward delay 0, hold 0
  Number of transitions to forwarding state: 1
  Link type is point-to-point by default
  PVST Simulation is disabled
  BPDU: sent 132, received 1
```

Examples: Detecting Unidirectional Link Failure

This example shows the spanning tree status when port `1/0/1 detail` has been configured to disable PVST+ simulation and the port is currently in the peer type inconsistent state:

```
Switch# show spanning-tree
VLAN0010
```

```
Spanning tree enabled protocol rstp
Root ID    Priority 32778
           Address 0002.172c.f400
           This bridge is the root
           Hello Time 2 sec Max Age 20 sec Forward Delay 15 sec
Bridge ID  Priority 32778 (priority 32768 sys-id-ext 10)
           Address 0002.172c.f400
           Hello Time 2 sec Max Age 20 sec Forward Delay 15 sec
           Aging Time 300
```

```
Interface      Role Sts Cost      Prio.Nbr Type
-----
Gi1/0/1        Desg BKN 4          128.270 P2p Dispute
```

This example shows the interface details when a dispute condition is detected:

```
Switch# show spanning-tree interface1/0/1 detail
Port 269 (GigabitEthernet1/0/1) of VLAN0002 is designated blocking (dispute)
  Port path cost 4, Port priority 128, Port Identifier 128.297.
  Designated root has priority 32769, address 0013.5f20.01c0
  Designated bridge has priority 32769, address 0013.5f20.01c0
  Designated port id is 128.297, designated path cost 0
  Timers: message age 0, forward delay 0, hold 0
  Number of transitions to forwarding state: 1
  Link type is point-to-point by default
  BPDUs: sent 132, received 1
```

Additional References for MSTP

Related Documents

Related Topic	Document Title
Layer 2 commands	<i>Catalyst 2960-XR Switch Layer 2 Command Reference</i>

Standards and RFCs

Standard/RFC	Title
None	—

MIBs

MIB	MIBs Link
All the supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
<p>The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies.</p> <p>To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds.</p> <p>Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.</p>	http://www.cisco.com/support

Feature Information for MSTP

Release	Modification
Cisco IOS Release 15.0(2)EX1	This feature was introduced.



CHAPTER 27

Configuring Optional Spanning-Tree Features

- [Restriction for Optional Spanning-Tree Features, on page 445](#)
- [Information About Optional Spanning-Tree Features, on page 445](#)
- [How to Configure Optional Spanning-Tree Features, on page 457](#)
- [Examples, on page 472](#)
- [Monitoring the Spanning-Tree Status, on page 475](#)
- [Additional References for Optional Spanning Tree Features, on page 475](#)
- [Feature Information for Optional Spanning-Tree Features, on page 476](#)

Restriction for Optional Spanning-Tree Features

- PortFast minimizes the time that interfaces must wait for spanning tree to converge, so it is effective only when used on interfaces connected to end stations. If you enable PortFast on an interface connecting to another switch, you risk creating a spanning-tree loop.

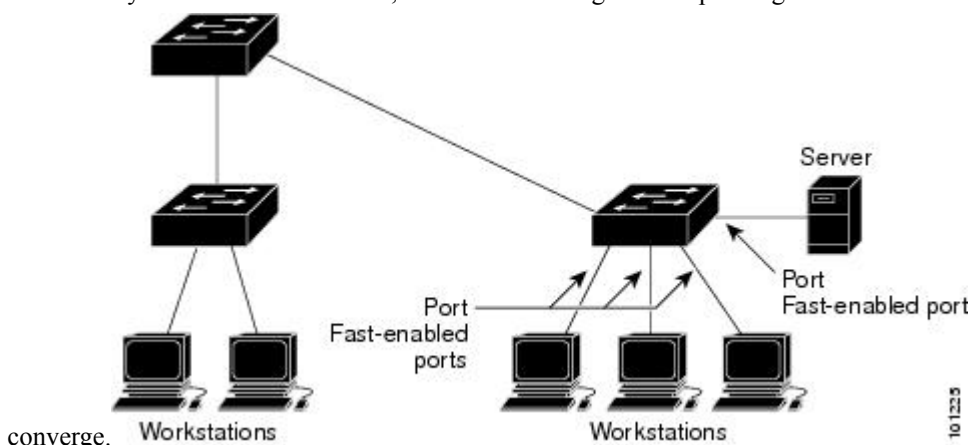
Information About Optional Spanning-Tree Features

PortFast

PortFast immediately brings an interface configured as an access or trunk port to the forwarding state from a blocking state, bypassing the listening and learning states.

Figure 33: PortFast-Enabled Interfaces

You can use PortFast on interfaces connected to a single workstation or server to allow those devices to immediately connect to the network, rather than waiting for the spanning tree to



Interfaces connected to a single workstation or server should not receive bridge protocol data units (BPDUs). An interface with PortFast enabled goes through the normal cycle of spanning-tree status changes when the switch is restarted.

You can enable this feature by enabling it on either the interface or on all nontrunking ports.

BPDU Guard

The Bridge Protocol Data Unit (BPDU) guard feature can be globally enabled on the switch or can be enabled per port, but the feature operates with some differences.

When you enable BPDU guard at the global level on PortFast edge-enabled ports, spanning tree shuts down ports that are in a PortFast edge-operational state if any BPDU is received on them. In a valid configuration, PortFast edge-enabled ports do not receive BPDUs. Receiving a BPDU on a Port Fast edge-enabled port means an invalid configuration, such as the connection of an unauthorized device, and the BPDU guard feature puts the port in the error-disabled state. When this happens, the switch shuts down the entire port on which the violation occurred.

When you enable BPDU guard at the interface level on any port without also enabling the PortFast edge feature, and the port receives a BPDU, it is put in the error-disabled state.

The BPDU guard feature provides a secure response to invalid configurations because you must manually put the interface back in service. Use the BPDU guard feature in a service-provider network to prevent an access port from participating in the spanning tree.

BPDU Filtering

The BPDU filtering feature can be globally enabled on the switch or can be enabled per interface, but the feature operates with some differences.

Enabling BPDU filtering on PortFast edge-enabled interfaces at the global level keeps those interfaces that are in a PortFast edge-operational state from sending or receiving BPDUs. The interfaces still send a few BPDUs at link-up before the switch begins to filter outbound BPDUs. You should globally enable BPDU filtering on a switch so that hosts connected to these interfaces do not receive BPDUs. If a BPDU is received

on a PortFast edge-enabled interface, the interface loses its PortFast edge-operational status, and BPDU filtering is disabled.

Enabling BPDU filtering on an interface without also enabling the PortFast edge feature keeps the interface from sending or receiving BPDUs.



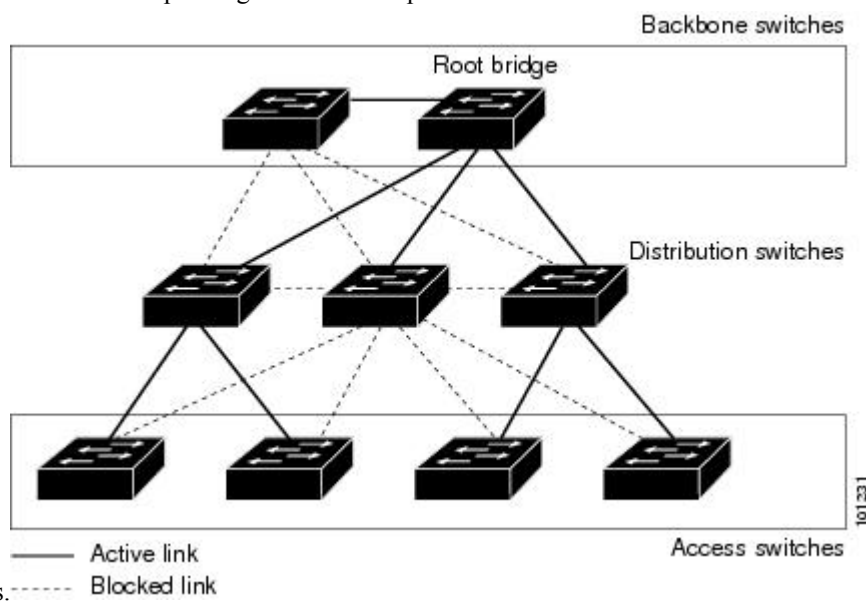
Caution Enabling BPDU filtering on an interface is the same as disabling spanning tree on it and can result in spanning-tree loops.

You can enable the BPDU filtering feature for the entire switch or for an interface.

UplinkFast

Figure 34: Switches in a Hierarchical Network

Switches in hierarchical networks can be grouped into backbone switches, distribution switches, and access switches. This complex network has distribution switches and access switches that each have at least one redundant link that spanning tree blocks to prevent loops.



If a switch loses connectivity, it begins using the alternate paths as soon as the spanning tree selects a new root port. You can accelerate the choice of a new root port when a link or switch fails or when the spanning tree reconfigures itself by enabling UplinkFast. The root port transitions to the forwarding state immediately without going through the listening and learning states, as it would with the normal spanning-tree procedures.

When the spanning tree reconfigures the new root port, other interfaces flood the network with multicast packets, one for each address that was learned on the interface. You can limit these bursts of multicast traffic by reducing the max-update-rate parameter (the default for this parameter is 150 packets per second). However, if you enter zero, station-learning frames are not generated, so the spanning-tree topology converges more slowly after a loss of connectivity.

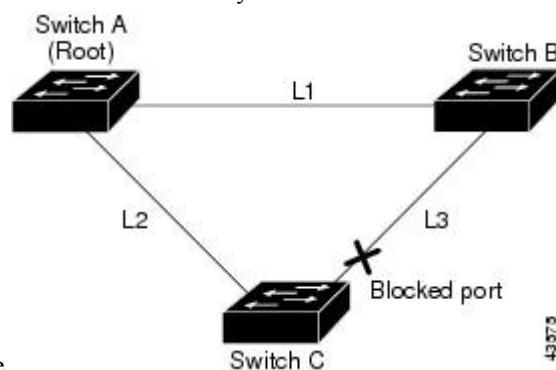


Note UplinkFast is most useful in wiring-closet switches at the access or edge of the network. It is not appropriate for backbone devices. This feature might not be useful for other types of applications.

UplinkFast provides fast convergence after a direct link failure and achieves load-balancing between redundant Layer 2 links using uplink groups. An uplink group is a set of Layer 2 interfaces (per VLAN), only one of which is forwarding at any given time. Specifically, an uplink group consists of the root port (which is forwarding) and a set of blocked ports, except for self-looping ports. The uplink group provides an alternate path in case the currently forwarding link fails.

Figure 35: UplinkFast Example Before Direct Link Failure

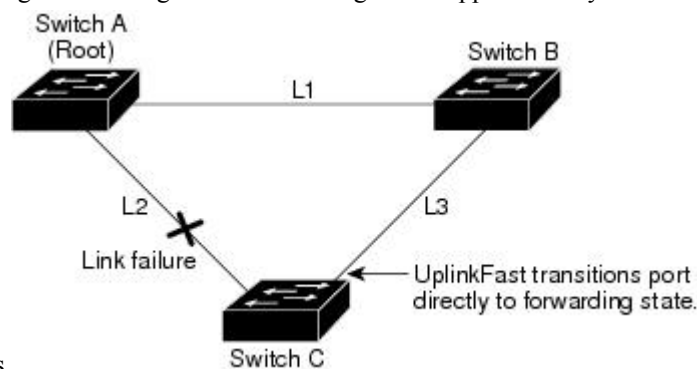
This topology has no link failures. Switch A, the root switch, is connected directly to Switch B over link L1 and to Switch C over link L2. The Layer 2 interface on Switch C that is connected directly to Switch B is in



a blocking state.

Figure 36: UplinkFast Example After Direct Link Failure

If Switch C detects a link failure on the currently active link L2 on the root port (a direct link failure), UplinkFast unblocks the blocked interface on Switch C and transitions it to the forwarding state without going through the listening and learning states. This change takes approximately 1 to



5 seconds.

Cross-Stack UplinkFast

Cross-Stack UplinkFast (CSUF) provides a fast spanning-tree transition (fast convergence in less than 1 second under normal network conditions) across a switch stack. During the fast transition, an alternate redundant link on the switch stack is placed in the forwarding state without causing temporary spanning-tree loops or loss of connectivity to the backbone. With this feature, you can have a redundant and resilient network in some configurations. CSUF is automatically enabled when you enable the UplinkFast feature.

CSUF might not provide a fast transition all the time; in these cases, the normal spanning-tree transition occurs, completing in 30 to 40 seconds. For more information, see [Related Topics](#).

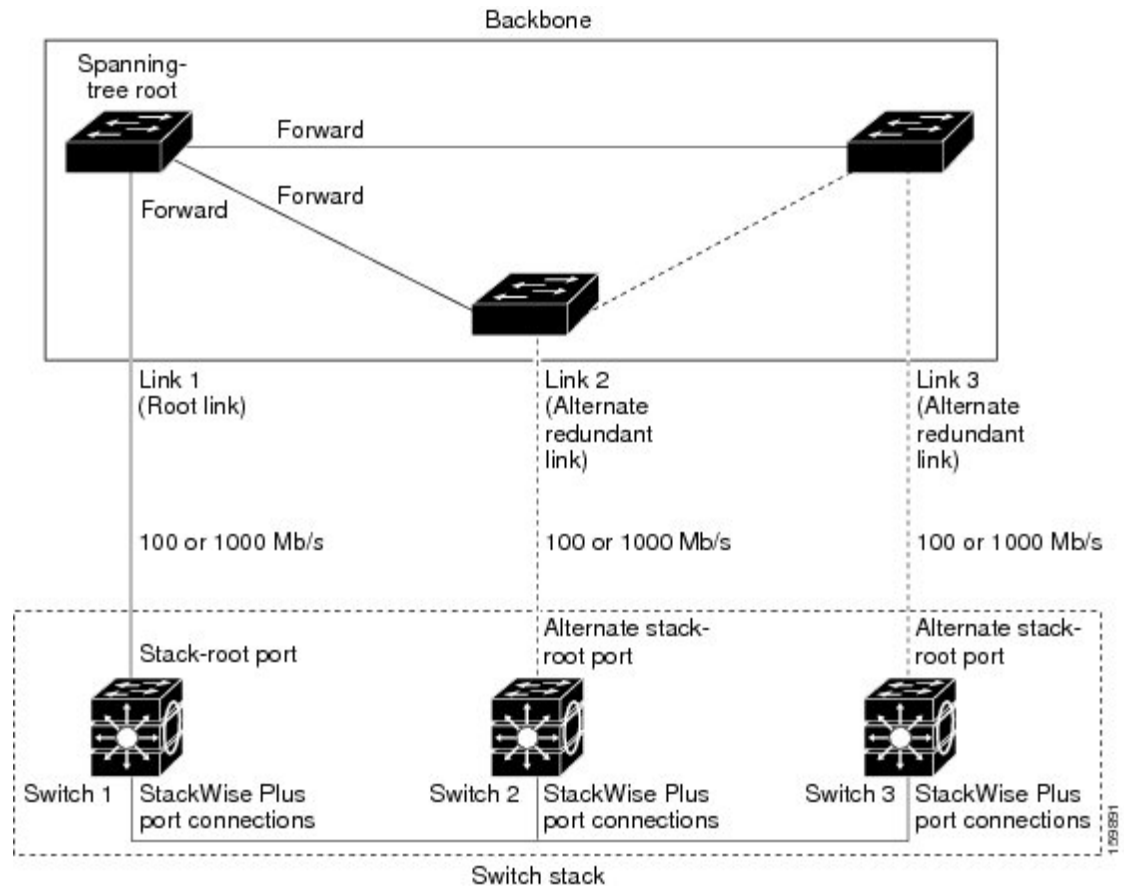
How Cross-Stack UplinkFast Works

Cross-Stack UplinkFast (CSUF) ensures that one link in the stack is elected as the path to the root.

Figure 37: Cross-Stack UplinkFast Topology

The stack-root port on Switch 1 provides the path to the root of the spanning tree. The alternate stack-root ports on Switches 2 and 3 can provide an alternate path to the spanning-tree root if the current stack-root switch fails or if its link to the spanning-tree root fails.

Link 1, the root link, is in the spanning-tree forwarding state. Links 2 and 3 are alternate redundant links that are in the spanning-tree blocking state. If Switch 1 fails, if its stack-root port fails, or if Link 1 fails, CSUF selects either the alternate stack-root port on Switch 2 or Switch 3 and puts it into the forwarding state in less than 1 second.



When certain link loss or spanning-tree events occur (described in the following topic), the Fast Uplink Transition Protocol uses the neighbor list to send fast-transition requests to stack members.

The switch sending the fast-transition request needs to do a fast transition to the forwarding state of a port that it has chosen as the root port, and it must obtain an acknowledgment from each stack switch before performing the fast transition.

Each switch in the stack decides if the sending switch is a better choice than itself to be the stack root of this spanning-tree instance by comparing the root, cost, and bridge ID. If the sending switch is the best choice as the stack root, each switch in the stack returns an acknowledgment; otherwise, it sends a fast-transition request. The sending switch then has not received acknowledgments from all stack switches.

When acknowledgments are received from all stack switches, the Fast Uplink Transition Protocol on the sending switch immediately transitions its alternate stack-root port to the forwarding state. If acknowledgments from all stack switches are not obtained by the sending switch, the normal spanning-tree transitions (blocking, listening, learning, and forwarding) take place, and the spanning-tree topology converges at its normal rate ($2 * \text{forward-delay time} + \text{max-age time}$).

The Fast Uplink Transition Protocol is implemented on a per-VLAN basis and affects only one spanning-tree instance at a time.

Events That Cause Fast Convergence

Depending on the network event or failure, the CSUF fast convergence might or might not occur.

Fast convergence (less than 1 second under normal network conditions) occurs under these circumstances:

- The stack-root port link fails.
If two switches in the stack have alternate paths to the root, only one of the switches performs the fast transition.
- The failed link, which connects the stack root to the spanning-tree root, recovers.
- A network reconfiguration causes a new stack-root switch to be selected.
- A network reconfiguration causes a new port on the current stack-root switch to be chosen as the stack-root port.



Note The fast transition might not occur if multiple events occur simultaneously. For example, if a stack member is powered off, and at the same time, the link connecting the stack root to the spanning-tree root comes back up, the normal spanning-tree convergence occurs.

Normal spanning-tree convergence (30 to 40 seconds) occurs under these conditions:

- The stack-root switch is powered off, or the software failed.
- The stack-root switch, which was powered off or failed, is powered on.
- A new switch, which might become the stack root, is added to the stack.

BackboneFast

BackboneFast detects indirect failures in the core of the backbone. BackboneFast is a complementary technology to the UplinkFast feature, which responds to failures on links directly connected to access switches.

BackboneFast optimizes the maximum-age timer, which controls the amount of time the switch stores protocol information received on an interface. When a switch receives an inferior BPDU from the designated port of another switch, the BPDU is a signal that the other switch might have lost its path to the root, and BackboneFast tries to find an alternate path to the root.

BackboneFast starts when a root port or blocked interface on a switch receives inferior BPDUs from its designated switch. An inferior BPDU identifies a switch that declares itself as both the root bridge and the designated switch. When a switch receives an inferior BPDU, it means that a link to which the switch is not directly connected (an indirect link) has failed (that is, the designated switch has lost its connection to the root switch). Under spanning-tree rules, the switch ignores inferior BPDUs for the maximum aging time (default is 20 seconds).

The switch tries to find if it has an alternate path to the root switch. If the inferior BPDU arrives on a blocked interface, the root port and other blocked interfaces on the switch become alternate paths to the root switch. (Self-looped ports are not considered alternate paths to the root switch.) If the inferior BPDU arrives on the root port, all blocked interfaces become alternate paths to the root switch. If the inferior BPDU arrives on the root port and there are no blocked interfaces, the switch assumes that it has lost connectivity to the root switch, causes the maximum aging time on the root port to expire, and becomes the root switch according to normal spanning-tree rules.

If the switch has alternate paths to the root switch, it uses these alternate paths to send a root link query (RLQ) request. The switch sends the RLQ request on all alternate paths to learn if any stack member has an alternate root to the root switch and waits for an RLQ reply from other switches in the network and in the stack. The switch sends the RLQ request on all alternate paths and waits for an RLQ reply from other switches in the network.

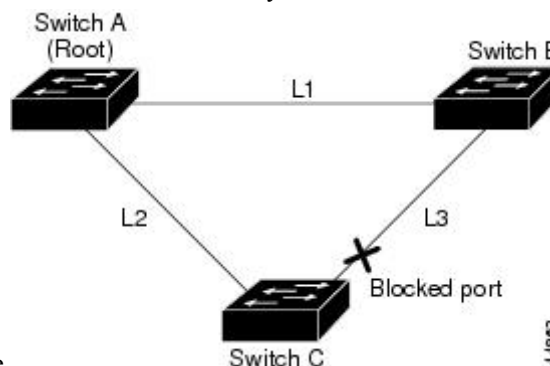
When a stack member receives an RLQ reply from a nonstack member on a blocked interface and the reply is destined for another nonstacked switch, it forwards the reply packet, regardless of the spanning-tree interface state.

When a stack member receives an RLQ reply from a nonstack member and the response is destined for the stack, the stack member forwards the reply so that all the other stack members receive it.

If the switch discovers that it still has an alternate path to the root, it expires the maximum aging time on the interface that received the inferior BPDU. If all the alternate paths to the root switch indicate that the switch has lost connectivity to the root switch, the switch expires the maximum aging time on the interface that received the RLQ reply. If one or more alternate paths can still connect to the root switch, the switch makes all interfaces on which it received an inferior BPDU its designated ports and moves them from the blocking state (if they were in the blocking state), through the listening and learning states, and into the forwarding state.

Figure 38: BackboneFast Example Before Indirect Link Failure

This is an example topology with no link failures. Switch A, the root switch, connects directly to Switch B over link L1 and to Switch C over link L2. The Layer 2 interface on Switch C that connects directly to Switch B



B is in the blocking state.

Figure 39: BackboneFast Example After Indirect Link Failure

If link L1 fails, Switch C cannot detect this failure because it is not connected directly to link L1. However, because Switch B is directly connected to the root switch over L1, it detects the failure, elects itself the root, and begins sending BPDUs to Switch C, identifying itself as the root. When Switch C receives the inferior BPDUs from Switch B, Switch C assumes that an indirect failure has occurred. At that point, BackboneFast allows the blocked interface on Switch C to move immediately to the listening state without waiting for the maximum aging time for the interface to expire. BackboneFast then transitions the Layer 2 interface on Switch C to the forwarding state, providing a path from Switch B to Switch A. The root-switch election takes approximately 30 seconds, twice the Forward Delay time if the default Forward Delay time of 15 seconds is set. BackboneFast reconfigures the topology to account for the failure of link

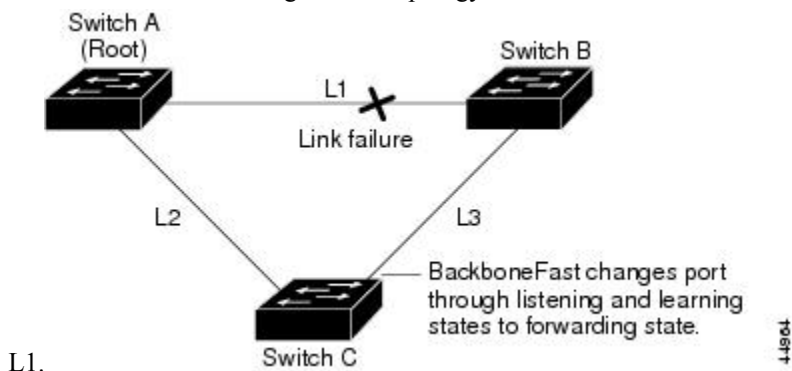
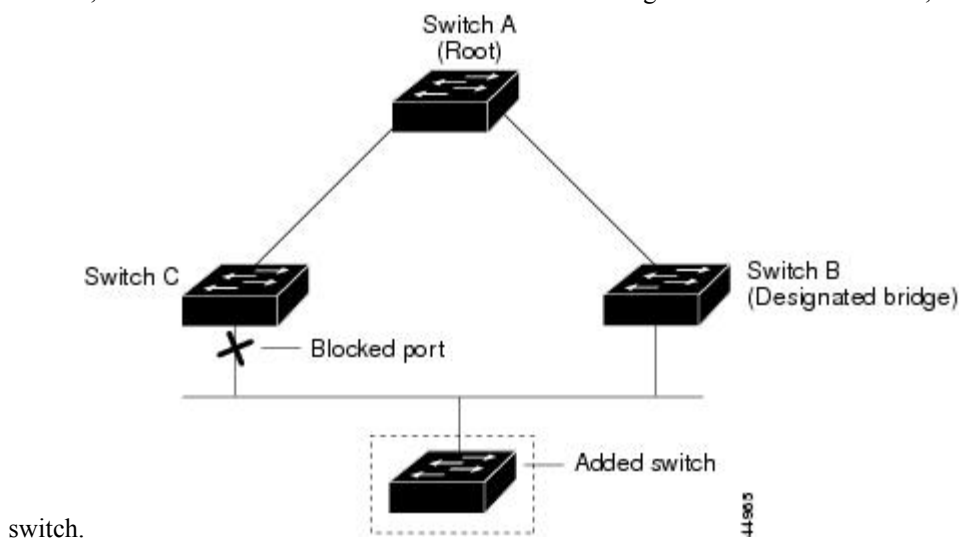


Figure 40: Adding a Switch in a Shared-Medium Topology

If a new switch is introduced into a shared-medium topology, BackboneFast is not activated because the inferior BPDUs did not come from the recognized designated switch (Switch B). The new switch begins sending inferior BPDUs that indicate it is the root switch. However, the other switches ignore these inferior BPDUs, and the new switch learns that Switch B is the designated switch to Switch A, the root



EtherChannel Guard

You can use EtherChannel guard to detect an EtherChannel misconfiguration between the switch and a connected device. A misconfiguration can occur if the switch interfaces are configured in an EtherChannel,

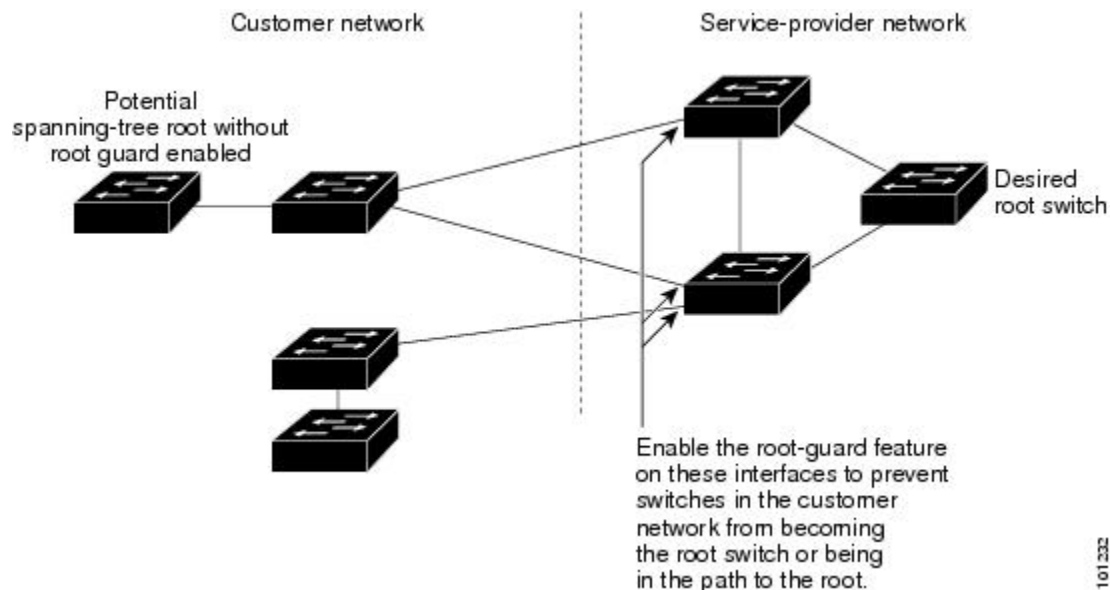
but the interfaces on the other device are not. A misconfiguration can also occur if the channel parameters are not the same at both ends of the EtherChannel.

If the switch detects a misconfiguration on the other device, EtherChannel guard places the switch interfaces in the error-disabled state, and displays an error message.

Root Guard

Figure 41: Root Guard in a Service-Provider Network

The Layer 2 network of a service provider (SP) can include many connections to switches that are not owned by the SP. In such a topology, the spanning tree can reconfigure itself and select a customer switch as the root switch. You can avoid this situation by enabling root guard on SP switch interfaces that connect to switches in your customer's network. If spanning-tree calculations cause an interface in the customer network to be selected as the root port, root guard then places the interface in the root-inconsistent (blocked) state to prevent the customer's switch from becoming the root switch or being in the path to the root.



If a switch outside the SP network becomes the root switch, the interface is blocked (root-inconsistent state), and spanning tree selects a new root switch. The customer's switch does not become the root switch and is not in the path to the root.

If the switch is operating in multiple spanning-tree (MST) mode, root guard forces the interface to be a designated port. If a boundary port is blocked in an internal spanning-tree (IST) instance because of root guard, the interface also is blocked in all MST instances. A boundary port is an interface that connects to a LAN, the designated switch of which is either an IEEE 802.1D switch or a switch with a different MST region configuration.

Root guard enabled on an interface applies to all the VLANs to which the interface belongs. VLANs can be grouped and mapped to an MST instance.



Caution Misuse of the root guard feature can cause a loss of connectivity.

Loop Guard

You can use loop guard to prevent alternate or root ports from becoming designated ports because of a failure that leads to a unidirectional link. This feature is most effective when it is enabled on the entire switched network. Loop guard prevents alternate and root ports from becoming designated ports, and spanning tree does not send BPDUs on root or alternate ports.

When the switch is operating in PVST+ or rapid-PVST+ mode, loop guard prevents alternate and root ports from becoming designated ports, and spanning tree does not send BPDUs on root or alternate ports.

When the switch is operating in MST mode, BPDUs are not sent on nonboundary ports only if the interface is blocked by loop guard in all MST instances. On a boundary port, loop guard blocks the interface in all MST instances.

STP PortFast Port Types

You can configure a spanning tree port as an edge port, a network port, or a normal port. A port can be in only one of these states at a given time. The default spanning tree port type is normal. You can configure the port type either globally or per interface.

Depending on the type of device to which the interface is connected, you can configure a spanning tree port as one of these port types:

- A PortFast edge port—is connected to a Layer 2 host. This can be either an access port or an edge trunk port (**portfast edge trunk**). This type of port interface immediately transitions to the forwarding state, bypassing the listening and learning states. Use PortFast edge on Layer 2 access ports connected to a single workstation or server to allow those devices to connect to the network immediately, rather than waiting for spanning tree to converge.

Even if the interface receives a bridge protocol data unit (BPDU), spanning tree does not place the port into the blocking state. Spanning tree sets the port's operating state to *non-port fast* even if the configured state remains *port fast edge* and starts participating in the topology change.



Note If you configure a port connected to a Layer 2 switch or bridge as an edge port, you might create a bridging loop.

- A PortFast network port—is connected only to a Layer 2 switch or bridge. Bridge Assurance is enabled only on PortFast network ports. For more information, refer to *Bridge Assurance*.



Note If you configure a port that is connected to a Layer 2 host as a spanning tree network port, the port will automatically move into the blocking state.

- A PortFast normal port—is the default type of spanning tree port.



Note Beginning with Cisco IOS Release 15.2(4)E, or IOS XE 3.8.0E, if you enter the **spanning-tree portfast** [trunk] command in the global or interface configuration mode, the system automatically saves it as **spanning-tree portfast edge** [trunk].

Bridge Assurance

You can use Bridge Assurance to help prevent looping conditions that are caused by unidirectional links (one-way traffic on a link or port), or a malfunction in a neighboring switch. Here a malfunction refers to a switch that is not able to run STP any more, while still forwarding traffic (a brain dead switch).

BPDUs are sent out on all operational network ports, including alternate and backup ports, for each hello time period. Bridge Assurance monitors the receipt of BPDUs on point-to-point links on all network ports. When a port does not receive BPDUs within the allotted hello time period, the port is put into a blocked state (the same as a port inconsistent state, which stops forwarding of frames). When the port resumes receipt of BPDUs, the port resumes normal spanning tree operations.

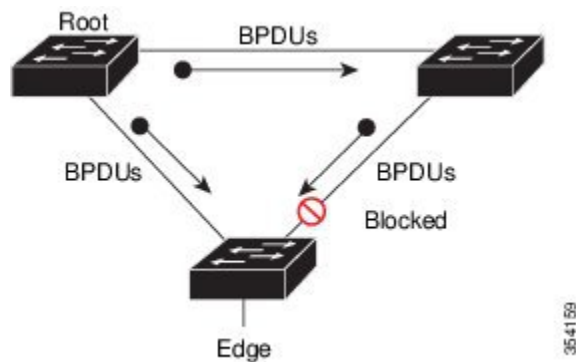


Note Only Rapid PVST+ and MST spanning tree protocols support Bridge Assurance. PVST+ does not support Bridge Assurance.

The following example shows how Bridge Assurance protects your network from bridging loops.

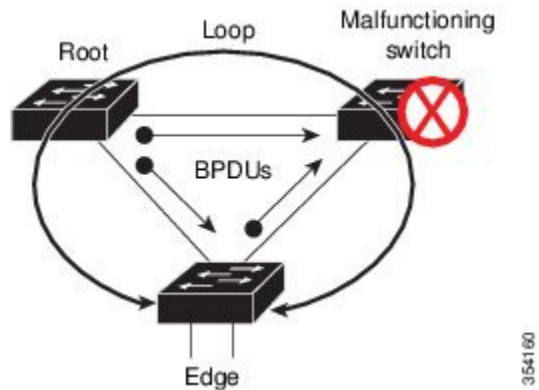
The following figure shows a network with normal STP topology.

Figure 42: Network with Normal STP Topology



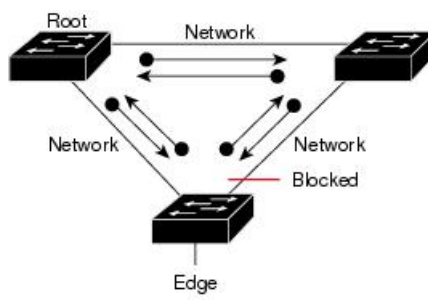
The following figure demonstrates a potential network problem when the device fails (brain dead) and Bridge Assurance is not enabled on the network.

Figure 43: Network Loop Due to a Malfunctioning Switch



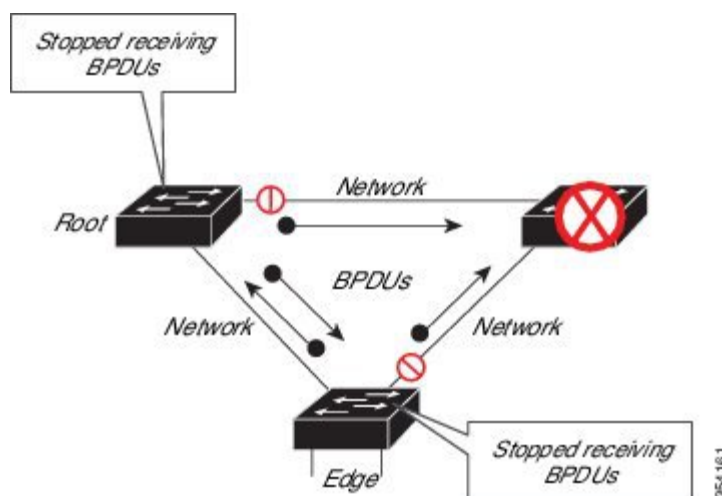
The following figure shows the network with Bridge Assurance enabled, and the STP topology progressing normally with bidirectional BDPUs issuing from every STP network port.

Figure 44: Network with STP Topology Running Bridge Assurance



The following figure shows how the potential network problem shown in figure *Network Loop Due to a Malfunctioning Switch* does not occur when you have Bridge Assurance enabled on your network.

Figure 45: Network Problem Averted with Bridge Assurance Enabled



The system generates syslog messages when a port is block and unblocked. The following sample output shows the log that is generated for each of these states:

BRIDGE_ASSURANCE_BLOCK

```
Sep 17 09:48:16.249 PDT: %SPANTREE-2-BRIDGE_ASSURANCE_BLOCK: Bridge Assurance blocking port
GigabitEthernet1/0/1 on VLAN0001.
```

BRIDGE_ASSURANCE_UNBLOCK

```
Sep 17 09:48:58.426 PDT: %SPANTREE-2-BRIDGE_ASSURANCE_UNBLOCK: Bridge Assurance unblocking
port GigabitEthernet1/0/1 on VLAN0001.
```

Follow these guidelines when enabling Bridge Assurance:

- It can only be enabled or disabled globally.
- It applies to all operational network ports, including alternate and backup ports.

- Only Rapid PVST+ and MST spanning tree protocols support Bridge Assurance. PVST+ does not support Bridge Assurance.
- For Bridge Assurance to work properly, it must be supported and configured on both ends of a point-to-point link. If the device on one side of the link has Bridge Assurance enabled and the device on the other side does not, the connecting port is blocked and in a Bridge Assurance inconsistent state. We recommend that you enable Bridge Assurance throughout your network.
- To enable Bridge Assurance on a port, BPDU filtering and BPDU Guard must be disabled.
- You can enable Bridge Assurance in conjunction with Loop Guard.
- You can enable Bridge Assurance in conjunction with Root Guard. The latter is designed to provide a way to enforce the root bridge placement in the network.

How to Configure Optional Spanning-Tree Features

Enabling PortFast

An interface with the PortFast feature enabled is moved directly to the spanning-tree forwarding state without waiting for the standard forward-time delay.

If you enable the voice VLAN feature, the PortFast feature is automatically enabled. When you disable voice VLAN, the PortFast feature is not automatically disabled.

You can enable this feature if your switch is running PVST+, Rapid PVST+, or MSTP.



Caution Use PortFast only when connecting a single end station to an access or trunk port. Enabling this feature on an interface connected to a switch or hub could prevent spanning tree from detecting and disabling loops in your network, which could cause broadcast storms and address-learning problems.

This procedure is optional.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **spanning-tree portfast** [**trunk**]
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example:	Enables privileged EXEC mode. • Enter your password if prompted.

	Command or Action	Purpose
	Device> enable	
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	interface <i>interface-id</i> Example: Device(config)# interface gigabitethernet 1/0/2	Specifies an interface to configure, and enters interface configuration mode.
Step 4	spanning-tree portfast [trunk] Example: Device(config-if)# spanning-tree portfast trunk	<p>Enables PortFast on an access port connected to a single workstation or server.</p> <p>By specifying the trunk keyword, you can enable PortFast on a trunk port.</p> <p>Note To enable PortFast on trunk ports, you must use the spanning-tree portfast trunk interface configuration command. The spanning-tree portfast command will not work on trunk ports.</p> <p>Make sure that there are no loops in the network between the trunk port and the workstation or server before you enable PortFast on a trunk port.</p> <p>By default, PortFast is disabled on all interfaces.</p>
Step 5	end Example: Device(config-if)# end	Returns to privileged EXEC mode.

What to do next

You can use the **spanning-tree portfast default** global configuration command to globally enable the PortFast feature on all nontrunking ports.

Enabling BPDU Guard

You can enable the BPDU guard feature if your switch is running PVST+, Rapid PVST+, or MSTP.



Caution Configure PortFast edge only on ports that connect to end stations; otherwise, an accidental topology loop could cause a data packet loop and disrupt switch and network operation.

This procedure is optional.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **spanning-tree portfast edge**
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	interface <i>interface-id</i> Example: Device(config)# interface gigabitethernet 1/0/2	Specifies the interface connected to an end station, and enters interface configuration mode.
Step 4	spanning-tree portfast edge Example: Device(config-if)# spanning-tree portfast edge	Enables the PortFast edge feature.
Step 5	end Example: Device(config-if)# end	Returns to privileged EXEC mode.

What to do next

To prevent the port from shutting down, you can use the **errdisable detect cause bpduguard shutdown vlan** global configuration command to shut down just the offending VLAN on the port where the violation occurred.

You also can use the **spanning-tree bpduguard enable** interface configuration command to enable BPDU guard on any port without also enabling the PortFast edge feature. When the port receives a BPDU, it is put it in the error-disabled state.

Enabling BPDU Filtering

You can also use the **spanning-tree bpdudfilter enable** interface configuration command to enable BPDU filtering on any interface without also enabling the PortFast edge feature. This command prevents the interface from sending or receiving BPDUs.



Caution Enabling BPDU filtering on an interface is the same as disabling spanning tree on it and can result in spanning-tree loops.

You can enable the BPDU filtering feature if your switch is running PVST+, Rapid PVST+, or MSTP.



Caution Configure PortFast edge only on interfaces that connect to end stations; otherwise, an accidental topology loop could cause a data packet loop and disrupt switch and network operation.

This procedure is optional.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **spanning-tree portfast edge bpdudfilter default**
4. **interface *interface-id***
5. **spanning-tree portfast edge**
6. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	spanning-tree portfast edge bpdudfilter default Example:	Globally enables BPDU filtering. By default, BPDU filtering is disabled.

	Command or Action	Purpose
	Device(config)# spanning-tree portfast edge bpdufilter default	
Step 4	interface <i>interface-id</i> Example: Device(config)# interface gigabitethernet 1/0/2	Specifies the interface connected to an end station, and enters interface configuration mode.
Step 5	spanning-tree portfast edge Example: Device(config-if)# spanning-tree portfast edge	Enables the PortFast edge feature on the specified interface.
Step 6	end Example: Device(config-if)# end	Returns to privileged EXEC mode.

Enabling UplinkFast for Use with Redundant Links



Note When you enable UplinkFast, it affects all VLANs on the switch or switch stack. You cannot configure UplinkFast on an individual VLAN.

You can configure the UplinkFast or the Cross-Stack UplinkFast (CSUF) feature for Rapid PVST+ or for the MSTP, but the feature remains disabled (inactive) until you change the spanning-tree mode to PVST+.

This procedure is optional. Follow these steps to enable UplinkFast and CSUF.

Before you begin

UplinkFast cannot be enabled on VLANs that have been configured with a switch priority. To enable UplinkFast on a VLAN with switch priority configured, first restore the switch priority on the VLAN to the default value using the **no spanning-tree vlan *vlan-id* priority** global configuration command.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **spanning-tree uplinkfast** [**max-update-rate** *pkts-per-second*]
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable	Enables privileged EXEC mode.

	Command or Action	Purpose
	Example: Device> enable	<ul style="list-style-type: none"> Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	spanning-tree uplinkfast [max-update-rate pkts-per-second] Example: Device(config)# spanning-tree uplinkfast max-update-rate 200	Enables UplinkFast. (Optional) For <i>pkts-per-second</i> , the range is 0 to 32000 packets per second; the default is 150. If you set the rate to 0, station-learning frames are not generated, and the spanning-tree topology converges more slowly after a loss of connectivity. When you enter this command, CSUF also is enabled on all nonstack port interfaces.
Step 4	end Example: Device(config)# end	Returns to privileged EXEC mode.

When UplinkFast is enabled, the switch priority of all VLANs is set to 49152. If you change the path cost to a value less than 3000 and you enable UplinkFast or UplinkFast is already enabled, the path cost of all interfaces and VLAN trunks is increased by 3000 (if you change the path cost to 3000 or above, the path cost is not altered). The changes to the switch priority and the path cost reduce the chance that a switch will become the root switch.

When UplinkFast is disabled, the switch priorities of all VLANs and path costs of all interfaces are set to default values if you did not modify them from their defaults.

When you enable the UplinkFast feature using these instructions, CSUF is automatically globally enabled on nonstack port interfaces.

Disabling UplinkFast

This procedure is optional.

Follow these steps to disable UplinkFast and Cross-Stack UplinkFast (CSUF).

Before you begin

UplinkFast must be enabled.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **no spanning-tree uplinkfast**
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	no spanning-tree uplinkfast Example: Device(config)# no spanning-tree uplinkfast	Disables UplinkFast and CSUF on the switch and all of its VLANs.
Step 4	end Example: Device(config)# end	Returns to privileged EXEC mode.

When UplinkFast is disabled, the switch priorities of all VLANs and path costs of all interfaces are set to default values if you did not modify them from their defaults.

When you disable the UplinkFast feature using these instructions, CSUF is automatically globally disabled on nonstack port interfaces.

Enabling BackboneFast

You can enable BackboneFast to detect indirect link failures and to start the spanning-tree reconfiguration sooner.

You can configure the BackboneFast feature for Rapid PVST+ or for the MSTP, but the feature remains disabled (inactive) until you change the spanning-tree mode to PVST+.

This procedure is optional. Follow these steps to enable BackboneFast on the switch.

Before you begin

If you use BackboneFast, you must enable it on all switches in the network. BackboneFast is not supported on Token Ring VLANs. This feature is supported for use with third-party switches.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **spanning-tree backbonefast**
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	spanning-tree backbonefast Example: Device(config)# spanning-tree backbonefast	Enables BackboneFast.
Step 4	end Example: Device(config)# end	Returns to privileged EXEC mode.

Enabling EtherChannel Guard

You can enable EtherChannel guard to detect an EtherChannel misconfiguration if your device is running PVST+, Rapid PVST+, or MSTP.

This procedure is optional.

Follow these steps to enable EtherChannel Guard on the device.

SUMMARY STEPS

1. **enable**

2. `configure terminal`
3. `spanning-tree etherchannel guard misconfig`
4. `end`

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p>enable</p> <p>Example:</p> <pre>Device> enable</pre>	<p>Enables privileged EXEC mode.</p> <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	<p>configure terminal</p> <p>Example:</p> <pre>Device# configure terminal</pre>	<p>Enters global configuration mode.</p>
Step 3	<p>spanning-tree etherchannel guard misconfig</p> <p>Example:</p> <pre>Device(config)# spanning-tree etherchannel guard misconfig</pre>	<p>Enables EtherChannel guard.</p>
Step 4	<p>end</p> <p>Example:</p> <pre>Device(config)# end</pre>	<p>Returns to privileged EXEC mode.</p>

What to do next

You can use the **show interfaces status err-disabled** privileged EXEC command to show which device ports are disabled because of an EtherChannel misconfiguration. On the remote device, you can enter the **show etherchannel summary** privileged EXEC command to verify the EtherChannel configuration.

After the configuration is corrected, enter the **shutdown** and **no shutdown** interface configuration commands on the port-channel interfaces that were misconfigured.

Enabling Root Guard

Root guard enabled on an interface applies to all the VLANs to which the interface belongs. Do not enable the root guard on interfaces to be used by the UplinkFast feature. With UplinkFast, the backup interfaces (in the blocked state) replace the root port in the case of a failure. However, if root guard is also enabled, all the backup interfaces used by the UplinkFast feature are placed in the root-inconsistent state (blocked) and are prevented from reaching the forwarding state.



Note You cannot enable both root guard and loop guard at the same time.

You can enable this feature if your switch is running PVST+, Rapid PVST+, or MSTP.

This procedure is optional.

Follow these steps to enable root guard on the switch.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **spanning-tree guard root**
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	interface <i>interface-id</i> Example: Device(config)# interface gigabitethernet 1/0/2	Specifies an interface to configure, and enters interface configuration mode.
Step 4	spanning-tree guard root Example: Device(config-if)# spanning-tree guard root	Enables root guard on the interface. By default, root guard is disabled on all interfaces.
Step 5	end Example: Device(config-if)# end	Returns to privileged EXEC mode.

Enabling Loop Guard

You can use loop guard to prevent alternate or root ports from becoming designated ports because of a failure that leads to a unidirectional link. This feature is most effective when it is configured on the entire switched network. Loop guard operates only on interfaces that are considered point-to-point by the spanning tree.



Note You cannot enable both loop guard and root guard at the same time.

You can enable this feature if your device is running PVST+, Rapid PVST+, or MSTP.

This procedure is optional. Follow these steps to enable loop guard on the device.

SUMMARY STEPS

1. Enter one of the following commands:
 - **show spanning-tree active**
 - **show spanning-tree mst**
2. **configure terminal**
3. **spanning-tree loopguard default**
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	Enter one of the following commands: <ul style="list-style-type: none"> • show spanning-tree active • show spanning-tree mst Example: Device# <code>show spanning-tree active</code> or Device# <code>show spanning-tree mst</code>	Verifies which interfaces are alternate or root ports.
Step 2	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode.
Step 3	spanning-tree loopguard default Example: Device(config)# <code>spanning-tree loopguard default</code>	Enables loop guard. By default, loop guard is disabled.

	Command or Action	Purpose
Step 4	end Example: Device(config)# end	Returns to privileged EXEC mode.

Enabling PortFast Port Types

This section describes the different steps to enable Portfast Port types.

Configuring the Default Port State Globally

To configure the default PortFast state, perform this task:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **spanning-tree portfast [edge | network | normal] default**
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	spanning-tree portfast [edge network normal] default Example: Device(config)# spanning-tree portfast default	Configures the default state for all interfaces on the switch. You have these options: <ul style="list-style-type: none"> • (Optional) edge—Configures all interfaces as edge ports. This assumes all ports are connected to hosts/servers. • (Optional) network—Configures all interfaces as spanning tree network ports. This assumes all ports are connected to switches and bridges. Bridge Assurance is enabled on all network ports by default.

	Command or Action	Purpose
		<ul style="list-style-type: none"> • (Optional) normal—Configures all interfaces normal spanning tree ports. These ports can be connected to any type of device. • default—The default port type is normal.
Step 4	end Example: Device(config)# end	Returns to privileged EXEC mode.

Configuring PortFast Edge on a Specified Interface

Interfaces configured as edge ports immediately transition to the forwarding state, without passing through the blocking or learning states, on linkup.



Note Because the purpose of this type of port is to minimize the time that access ports must wait for spanning tree to converge, it is most effective when used on access ports. If you enable PortFast edge on a port connecting to another switch, you risk creating a spanning tree loop.

To configure an edge port on a specified interface, perform this task:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id* | **port-channel** *port_channel_number*
4. **spanning-tree portfast edge** [**trunk**]
5. **end**
6. **show running interface** *interface-id* | **port-channel** *port_channel_number*

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.

	Command or Action	Purpose
Step 3	interface <i>interface-id</i> port-channel <i>port_channel_number</i> Example: Device(config)# interface gigabitethernet 1/0/1 port-channel <i>port_channel_number</i>	Specifies an interface to configure.
Step 4	spanning-tree portfast edge [trunk] Example: Device(config-if)# spanning-tree portfast trunk	Enables edge behavior on a Layer 2 access port connected to an end workstation or server. <ul style="list-style-type: none"> • (Optional) trunk—Enables edge behavior on a trunk port. Use this keyword if the link is a trunk. Use this command only on ports that are connected to end host devices that terminate VLANs and from which the port should never receive STP BPDUs. Such end host devices include workstations, servers, and ports on routers that are not configured to support bridging. • Use the no version of the command to disable PortFast edge.
Step 5	end Example: Device(config-if)# end	Exits configuration mode.
Step 6	show running interface <i>interface-id</i> port-channel <i>port_channel_number</i> Example: Device# show running interface gigabitethernet 1/0/1 port-channel <i>port_channel_number</i>	Verifies the configuration.

Configuring a PortFast Network Port on a Specified Interface

Ports that are connected to Layer 2 switches and bridges can be configured as network ports.



Note Bridge Assurance is enabled only on PortFast network ports. For more information, refer to *Bridge Assurance*.

To configure a port as a network port, perform this task.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id* | **port-channel** *port_channel_number*
4. **spanning-tree portfast network**
5. **end**

6. `show running interface interface-id | port-channel port_channel_number`

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	interface interface-id port-channel port_channel_number Example: Device(config)# interface gigabitethernet 1/0/1 port-channel port_channel_number	Specifies an interface to configure.
Step 4	spanning-tree portfast network Example: Device(config-if)# spanning-tree portfast network	Enables edge behavior on a Layer 2 access port connected to an end workstation or server. <ul style="list-style-type: none"> • Configures the port as a network port. If you have enabled Bridge Assurance globally, it automatically runs on a spanning tree network port. • Use the no version of the command to disable PortFast.
Step 5	end Example: Device(config-if)# end	Exits configuration mode.
Step 6	show running interface interface-id port-channel port_channel_number Example: Device# show running interface gigabitethernet 1/0/1 port-channel port_channel_number	Verifies the configuration.

Enabling Bridge Assurance

To configure the Bridge Assurance, perform the steps given below:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **spanning-tree bridge assurance**
4. **end**
5. **show spanning-tree summary**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	spanning-tree bridge assurance Example: Device(config)# spanning-tree bridge assurance	Enables Bridge Assurance on all network ports on the switch. Bridge Assurance is enabled by default. Use the no version of the command to disable the feature. Disabling Bridge Assurance causes all configured network ports to behave as normal spanning tree ports.
Step 4	end Example: Device(config)# end	Returns to privileged EXEC mode.
Step 5	show spanning-tree summary Example: Device# show spanning-tree summary	Displays spanning tree information and shows if Bridge Assurance is enabled.

Examples

Examples: Configuring PortFast Edge on a Specified Interface

This example shows how to enable edge behavior on GigabitEthernet interface **1/0/1**:

```
Switch# configure terminal
Switch(config)# interface gigabitethernet1/0/1
Switch(config-if)# spanning-tree portfast edge
Switch(config-if)# end
Switch#
```

This example shows how to verify the configuration:

```
Switch# show running-config interface gigabitethernet1/0/1
Building configuration...
Current configuration:
!
interface GigabitEthernet1/0/1
no ip address
switchport
switchport access vlan 200
switchport mode access
spanning-tree portfast edge
end
```

This example shows how you can display that port GigabitEthernet 1/0/1 is currently in the edge state:

```
Switch# show spanning-tree vlan 200
VLAN0200
Spanning tree enabled protocol rstp
Root ID Priority 2
Address 001b.2a68.5fc0
Cost 3
Port 125 (GigabitEthernet5/9)
Hello Time 2 sec Max Age 20 sec Forward Delay 15 sec
Bridge ID Priority 2 (priority 0 sys-id-ext 2)
Address 7010.5c9c.5200
Hello Time 2 sec Max Age 20 sec Forward Delay 15 sec
Aging Time 0 sec
Interface Role Sts Cost Prio.Nbr Type
-----
Gil/0/1 Desg FWD 4 128.1 P2p Edge
```

Examples: Configuring a PortFast Network Port on a Specified Interface

This example shows how to configure GigabitEthernet interface 1/0/1 as a network port:

```
Switch# configure terminal
Switch(config)# interface gigabitethernet1/0/1
Switch(config-if)# spanning-tree portfast network
Switch(config-if)# end
Switch#
```

This example shows how to verify the configuration:

```
Switch# show running-config interface gigabitethernet1/0/1
Building configuration...
Current configuration:
!
interface GigabitEthernet1/0/1
no ip address
switchport
switchport access vlan 200
switchport mode access
spanning-tree portfast network
end
```

This example shows the output for show spanning-tree vlan

```
Switch# show spanning-tree vlan
Sep 17 09:51:36.370 PDT: %SYS-5-CONFIG_I: Configured from console by console2

VLAN0002
  Spanning tree enabled protocol rstp
  Root ID    Priority    2
            Address    7010.5c9c.5200
            This bridge is the root
            Hello Time 2 sec  Max Age 20 sec  Forward Delay 15 sec

  Bridge ID  Priority    2          (priority 0 sys-id-ext 2)
            Address    7010.5c9c.5200
            Hello Time 2 sec  Max Age 20 sec  Forward Delay 15 sec
            Aging Time 0      sec

Interface                Role Sts Cost          Prio.Nbr Type
-----
Gi1/0/1                   Desg FWD 4             128.1    P2p Edge
Po4                       Desg FWD 3             128.480  P2p Network
Gi4/0/1                   Desg FWD 4             128.169  P2p Edge
Gi4/0/47                  Desg FWD 4             128.215  P2p Network

Switch#
```

Example: Configuring Bridge Assurance

This output shows port GigabitEthernet 1/0/1 has been configured as a network port and it is currently in the Bridge Assurance inconsistent state.



Note The output shows the port type as network and *BA_Inc, indicating that the port is in an inconsistent state.

```
Switch# show spanning-tree
VLAN0010
  Spanning tree enabled protocol rstp
  Root ID Priority 32778
  Address 0002.172c.f400
  This bridge is the root
  Hello Time 2 sec Max Age 20 sec Forward Delay 15 sec
  Bridge ID Priority 32778 (priority 32768 sys-id-ext 10)
  Address 0002.172c.f400
  Hello Time 2 sec Max Age 20 sec Forward Delay 15 sec
  Aging Time 300
  Interface Role Sts Cost Prio. Nbr Type
  -----
Gi1/0/1    Desg BKN*4 128.270 Network, P2p *BA_Inc
```

The example shows the output for show spanning-tree summary.

```
Switch#sh spanning-tree summary
Switch is in rapid-pvst mode
Root bridge for: VLAN0001-VLAN0002, VLAN0128
EtherChannel misconfig guard          is enabled
Extended system ID                    is enabled
Portfast Default                      is network
Portfast Edge BPDU Guard Default      is disabled
Portfast Edge BPDU Filter Default     is disabled
```

```

Loopguard Default           is enabled
PVST Simulation Default     is enabled but inactive in rapid-pvst mode
Bridge Assurance           is enabled
UplinkFast                 is disabled
BackboneFast               is disabled
Configured Pathcost method used is short

```

```

Name                Blocking Listening Learning Forwarding STP Active
-----
VLAN0001            0          0          0          5          5
VLAN0002            0          0          0          4          4
VLAN0128            0          0          0          4          4
-----
3 vlans             0          0          0          13         13

```

```
Switch#
```

Monitoring the Spanning-Tree Status

Table 52: Commands for Monitoring the Spanning-Tree Status

Command	Purpose
show spanning-tree active	Displays spanning-tree information on active interfaces only.
show spanning-tree detail	Displays a detailed summary of interface information.
show spanning-tree interface <i>interface-id</i>	Displays spanning-tree information for the specified interface.
show spanning-tree mst interface <i>interface-id</i>	Displays MST information for the specified interface.
show spanning-tree summary [totals]	Displays a summary of interface states or displays the total lines of the spanning-tree state section.
show spanning-tree mst interface <i>interface-id</i> portfast edge	Displays spanning-tree portfast information for the specified interface.

Additional References for Optional Spanning Tree Features

Related Documents

Related Topic	Document Title
Layer 2 commands	Catalyst 2960-XR Switch Layer 2 Command Reference

Standards and RFCs

Standard/RFC	Title
None	—

MIBs

MIB	MIBs Link
All the supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
<p>The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies.</p> <p>To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds.</p> <p>Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.</p>	http://www.cisco.com/support

Feature Information for Optional Spanning-Tree Features

Release	Modification
Cisco IOS Release 15.0(2)EX1	This feature was introduced.



CHAPTER 28

Configuring EtherChannels

- [Restrictions for EtherChannels, on page 477](#)
- [Information About EtherChannels, on page 477](#)
- [How to Configure EtherChannels, on page 491](#)
- [Monitoring EtherChannel, PAgP, and LACP Status, on page 508](#)
- [Configuration Examples for Configuring EtherChannels, on page 509](#)
- [Additional References for EtherChannels, on page 513](#)
- [Feature Information for EtherChannels, on page 514](#)

Restrictions for EtherChannels

- Layer 3 EtherChannels are not supported if the switch is running the LAN Base feature set.
- All ports in an EtherChannel must be assigned to the same VLAN or they must be configured as trunk ports.
- When the ports in an EtherChannel are configured as trunk ports, all the ports must be configured with the same mode (either Inter-Switch Link [ISL] or IEEE 802.1Q).
- Port Aggregation Protocol (PAgP) can be enabled only in single-switch EtherChannel configurations; PAgP cannot be enabled on cross-stack EtherChannels.

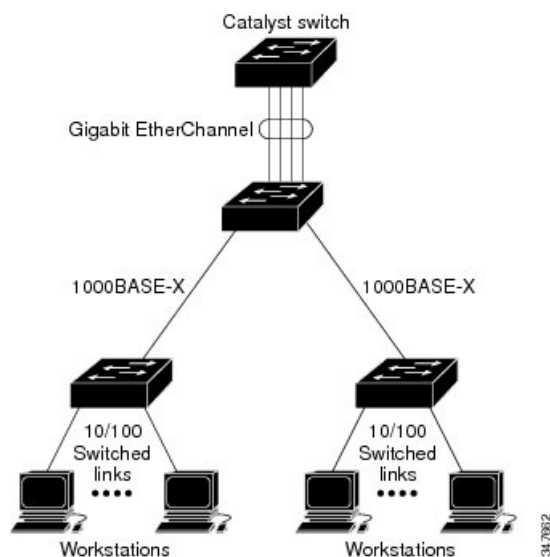
Information About EtherChannels

EtherChannel Overview

EtherChannel provides fault-tolerant high-speed links between switches, routers, and servers. You can use the EtherChannel to increase the bandwidth between the wiring closets and the data center, and you can deploy it anywhere in the network where bottlenecks are likely to occur. EtherChannel provides automatic recovery for the loss of a link by redistributing the load across the remaining links. If a link fails, EtherChannel redirects traffic from the failed link to the remaining links in the channel without intervention.

An EtherChannel consists of individual Ethernet links bundled into a single logical link.

Figure 46: Typical EtherChannel Configuration



Each EtherChannel can consist of up to eight compatibly configured Ethernet ports.

The LAN Base feature set supports up to 24 EtherChannels.

The IP Lite feature set supports up to 48 EtherChannels.

EtherChannel Modes

You can configure an EtherChannel in one of these modes: Port Aggregation Protocol (PAgP), Link Aggregation Control Protocol (LACP), or On. Configure both ends of the EtherChannel in the same mode:

- When you configure one end of an EtherChannel in either PAgP or LACP mode, the system negotiates with the other end of the channel to determine which ports should become active. If the remote port cannot negotiate an EtherChannel, the local port is put into an independent state and continues to carry data traffic as would any other single link. The port configuration does not change, but the port does not participate in the EtherChannel.
- When you configure an EtherChannel in the **on** mode, no negotiations take place. The switch forces all compatible ports to become active in the EtherChannel. The other end of the channel (on the other switch) must also be configured in the **on** mode; otherwise, packet loss can occur.

EtherChannel on Devices

You can create an EtherChannel on a device, on a single device in the stack, or on multiple devices in the stack (known as cross-stack EtherChannel).

Figure 47: Single-Switch EtherChannel

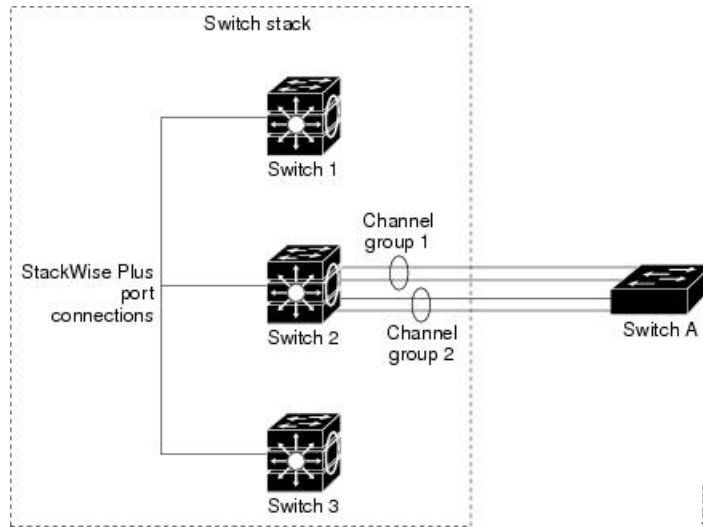
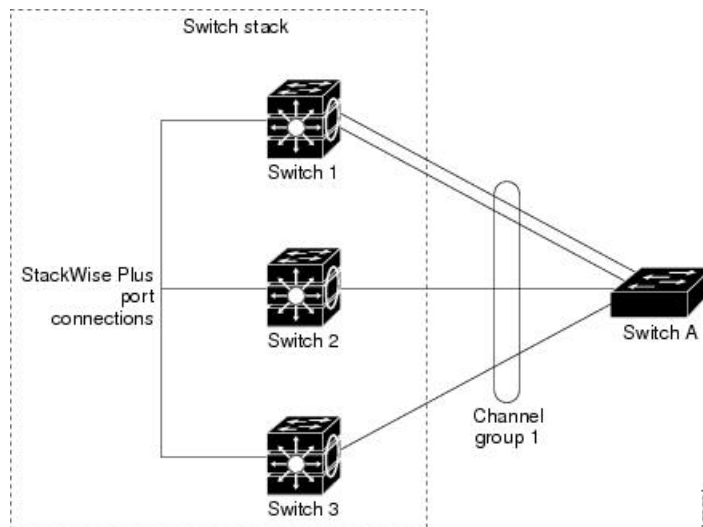


Figure 48: Cross-Stack EtherChannel



EtherChannel Link Failover

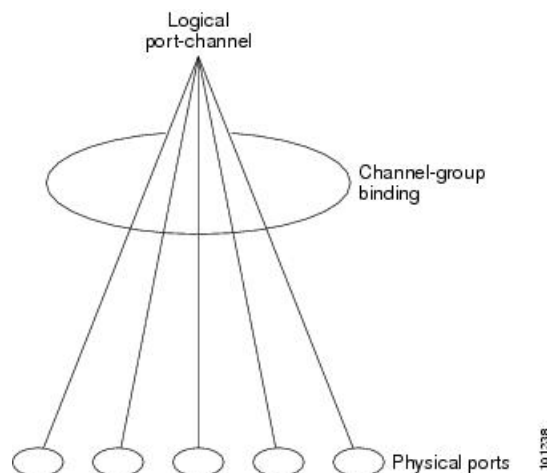
If a link within an EtherChannel fails, traffic previously carried over that failed link moves to the remaining links within the EtherChannel. If traps are enabled on the switch, a trap is sent for a failure that identifies the switch, the EtherChannel, and the failed link. Inbound broadcast and multicast packets on one link in an EtherChannel are blocked from returning on any other link of the EtherChannel.

Channel Groups and Port-Channel Interfaces

An EtherChannel comprises a channel group and a port-channel interface. The channel group binds physical ports to the port-channel interface. Configuration changes applied to the port-channel interface apply to all the physical ports bound together in the channel group.

Figure 49: Relationship of Physical Ports, Channel Group and Port-Channel Interface

The **channel-group** command binds the physical port and the port-channel interface together. Each EtherChannel has a port-channel logical interface numbered from 1 to 48. This port-channel interface number corresponds to the one specified with the **channel-group** interface configuration command.



- With Layer 2 ports, use the **channel-group** interface configuration command to dynamically create the port-channel interface.

You also can use the **interface port-channel** *port-channel-number* global configuration command to manually create the port-channel interface, but then you must use the **channel-group** *channel-group-number* command to bind the logical interface to a physical port. The *channel-group-number* can be the same as the *port-channel-number*, or you can use a new number. If you use a new number, the **channel-group** command dynamically creates a new port channel.

Port Aggregation Protocol

The Port Aggregation Protocol (PAgP) is a Cisco-proprietary protocol that can be run only on Cisco devices and on those devices licensed by vendors to support PAgP. PAgP facilitates the automatic creation of EtherChannels by exchanging PAgP packets between Ethernet ports.

By using PAgP, the device or device stack learns the identity of partners capable of supporting PAgP and the capabilities of each port. It then dynamically groups similarly configured ports (on a single device in the stack) into a single logical link (channel or aggregate port). Similarly configured ports are grouped based on hardware, administrative, and port parameter constraints. For example, PAgP groups the ports with the same speed, duplex mode, native VLAN, VLAN range, and trunking status and type. After grouping the links into an EtherChannel, PAgP adds the group to the spanning tree as a single device port.

PAgP Modes

PAgP modes specify whether a port can send PAgP packets, which start PAgP negotiations, or only respond to PAgP packets received.

Table 53: EtherChannel PAgP Modes

Mode	Description
auto	Places a port into a passive negotiating state, in which the port responds to PAgP packets it receives but does not start PAgP packet negotiation. This setting minimizes the transmission of PAgP packets. This mode is not supported when the EtherChannel members are from different switches in the switch stack (cross-stack EtherChannel).
desirable	Places a port into an active negotiating state, in which the port starts negotiations with other ports by sending PAgP packets. This mode is not supported when the EtherChannel members are from different switches in the switch stack (cross-stack EtherChannel).

Switch ports exchange PAgP packets only with partner ports configured in the **auto** or **desirable** modes. Ports configured in the **on** mode do not exchange PAgP packets.

Both the **auto** and **desirable** modes enable ports to negotiate with partner ports to form an EtherChannel based on criteria such as port speed, and for Layer 2 EtherChannels, based on trunk state and VLAN numbers.

Ports can form an EtherChannel when they are in different PAgP modes as long as the modes are compatible. For example:

- A port in the **desirable** mode can form an EtherChannel with another port that is in the **desirable** or **auto** mode.
- A port in the **auto** mode can form an EtherChannel with another port in the **desirable** mode.

A port in the **auto** mode cannot form an EtherChannel with another port that is also in the **auto** mode because neither port starts PAgP negotiation.

Silent Mode

If your switch is connected to a partner that is PAgP-capable, you can configure the switch port for nonsilent operation by using the **non-silent** keyword. If you do not specify **non-silent** with the **auto** or **desirable** mode, silent mode is assumed.

Use the silent mode when the switch is connected to a device that is not PAgP-capable and seldom, if ever, sends packets. An example of a silent partner is a file server or a packet analyzer that is not generating traffic. In this case, running PAgP on a physical port connected to a silent partner prevents that switch port from ever becoming operational. However, the silent setting allows PAgP to operate, to attach the port to a channel group, and to use the port for transmission.

PAgP Learn Method and Priority

Network devices are classified as PAgP physical learners or aggregate-port learners. A device is a physical learner if it learns addresses by physical ports and directs transmissions based on that knowledge. A device is an aggregate-port learner if it learns addresses by aggregate (logical) ports. The learn method must be configured the same at both ends of the link.

When a device and its partner are both aggregate-port learners, they learn the address on the logical port-channel. The device sends packets to the source by using any of the ports in the EtherChannel. With aggregate-port learning, it is not important on which physical port the packet arrives.

PAGP cannot automatically detect when the partner device is a physical learner and when the local device is an aggregate-port learner. Therefore, you must manually set the learning method on the local device to learn addresses by physical ports. You also must set the load-distribution method to source-based distribution, so that any given source MAC address is always sent on the same physical port.

You also can configure a single port within the group for all transmissions and use other ports for hot-standby. The unused ports in the group can be swapped into operation in just a few seconds if the selected single port loses hardware-signal detection. You can configure which port is always selected for packet transmission by changing its priority with the **pagp port-priority** interface configuration command. The higher the priority, the more likely that the port will be selected.



Note The device supports address learning only on aggregate ports even though the **physical-port** keyword is provided in the CLI. The **pagp learn-method** command and the **pagp port-priority** command have no effect on the device hardware, but they are required for PAGP interoperability with devices that only support address learning by physical ports, such as the Catalyst 1900 switch.

When the link partner of the device is a physical learner, we recommend that you configure the device as a physical-port learner by using the **pagp learn-method physical-port** interface configuration command. Set the load-distribution method based on the source MAC address by using the **port-channel load-balance src-mac** global configuration command. The device then sends packets to the physical learner using the same port in the EtherChannel from which it learned the source address. Only use the **pagp learn-method** command in this situation.

PAGP Interaction with Virtual Switches and Dual-Active Detection

A virtual switch can be two or more core switches connected by virtual switch links (VSLs) that carry control and data traffic between them. One of the switches is in active mode. The others are in standby mode. For redundancy, remote switches are connected to the virtual switch by remote satellite links (RSLs).

If the VSL between two switches fails, one switch does not know the status of the other. Both switches could change to the active mode, causing a *dual-active situation* in the network with duplicate configurations (including duplicate IP addresses and bridge identifiers). The network might go down.

To prevent a dual-active situation, the core switches send PAGP protocol data units (PDUs) through the RSLs to the remote switches. The PAGP PDUs identify the active switch, and the remote switches forward the PDUs to core switches so that the core switches are in sync. If the active switch fails or resets, the standby switch takes over as the active switch. If the VSL goes down, one core switch knows the status of the other and does not change its state.

PAGP Interaction with Other Features

The Dynamic Trunking Protocol (DTP) and the Cisco Discovery Protocol (CDP) send and receive packets over the physical ports in the EtherChannel. Trunk ports send and receive PAGP protocol data units (PDUs) on the lowest numbered VLAN.

In Layer 2 EtherChannels, the first port in the channel that comes up provides its MAC address to the EtherChannel. If this port is removed from the bundle, one of the remaining ports in the bundle provides its MAC address to the EtherChannel.

PAGP sends and receives PAGP PDUs only from ports that are up and have PAGP enabled for the auto or desirable mode.

Link Aggregation Control Protocol

The LACP is defined in IEEE 802.3ad and enables Cisco devices to manage Ethernet channels between devices that conform to the IEEE 802.3ad protocol. LACP facilitates the automatic creation of EtherChannels by exchanging LACP packets between Ethernet ports.

By using LACP, the device or device stack learns the identity of partners capable of supporting LACP and the capabilities of each port. It then dynamically groups similarly configured ports into a single logical link (channel or aggregate port). Similarly configured ports are grouped based on hardware, administrative, and port parameter constraints. For example, LACP groups the ports with the same speed, duplex mode, native VLAN, VLAN range, and trunking status and type. After grouping the links into an EtherChannel, LACP adds the group to the spanning tree as a single device port.

The independent mode behavior of ports in a port channel is changed. With CSCtn96950, by default, standalone mode is enabled. When no response is received from an LACP peer, ports in the port channel are moved to suspended state.

LACP Modes

LACP modes specify whether a port can send LACP packets or only receive LACP packets.

Table 54: EtherChannel LACP Modes

Mode	Description
active	Places a port into an active negotiating state in which the port starts negotiations with other ports by sending LACP packets.
passive	Places a port into a passive negotiating state in which the port responds to LACP packets that it receives, but does not start LACP packet negotiation. This setting minimizes the transmission of LACP packets.

Both the **active** and **passive LACP** modes enable ports to negotiate with partner ports to an EtherChannel based on criteria such as port speed, and for Layer 2 EtherChannels, based on trunk state and VLAN numbers.

Ports can form an EtherChannel when they are in different LACP modes as long as the modes are compatible. For example:

- A port in the **active** mode can form an EtherChannel with another port that is in the **active** or **passive** mode.
- A port in the **passive** mode cannot form an EtherChannel with another port that is also in the **passive** mode because neither port starts LACP negotiation.

LACP Interaction with Other Features

The DTP and the CDP send and receive packets over the physical ports in the EtherChannel. Trunk ports send and receive LACP PDUs on the lowest numbered VLAN.

In Layer 2 EtherChannels, the first port in the channel that comes up provides its MAC address to the EtherChannel. If this port is removed from the bundle, one of the remaining ports in the bundle provides its MAC address to the EtherChannel.

LACP sends and receives LACP PDUs only from ports that are up and have LACP enabled for the active or passive mode.

EtherChannel On Mode

EtherChannel **on** mode can be used to manually configure an EtherChannel. The **on** mode forces a port to join an EtherChannel without negotiations. The **on** mode can be useful if the remote device does not support PAgP or LACP. In the **on** mode, a usable EtherChannel exists only when the devices at both ends of the link are configured in the **on** mode.

Ports that are configured in the **on** mode in the same channel group must have compatible port characteristics, such as speed and duplex. Ports that are not compatible are suspended, even though they are configured in the **on** mode.



Caution You should use care when using the **on** mode. This is a manual configuration, and ports on both ends of the EtherChannel must have the same configuration. If the group is misconfigured, packet loss or spanning-tree loops can occur.

Load-Balancing and Forwarding Methods

EtherChannel balances the traffic load across the links in a channel by reducing part of the binary pattern formed from the addresses in the frame to a numerical value that selects one of the links in the channel. You can specify one of several different load-balancing modes, including load distribution based on MAC addresses, IP addresses, source addresses, destination addresses, or both source and destination addresses. The selected mode applies to all EtherChannels configured on the device.



Note Layer 3 Equal-cost multi path (ECMP) load balancing is based on source IP address, destination IP address, source port, destination port, and layer 4 protocol. Fragmented packets will be treated on two different links based on the algorithm calculated using these parameters. Any changes in one of these parameters will result in load balancing.

You configure the load-balancing and forwarding method by using the **port-channel load-balance** global configuration command.

MAC Address Forwarding

With source-MAC address forwarding, when packets are forwarded to an EtherChannel, they are distributed across the ports in the channel based on the source-MAC address of the incoming packet. Therefore, to provide load-balancing, packets from different hosts use different ports in the channel, but packets from the same host use the same port in the channel.

With destination-MAC address forwarding, when packets are forwarded to an EtherChannel, they are distributed across the ports in the channel based on the destination host's MAC address of the incoming packet. Therefore,

packets to the same destination are forwarded over the same port, and packets to a different destination are sent on a different port in the channel.

With source-and-destination MAC address forwarding, when packets are forwarded to an EtherChannel, they are distributed across the ports in the channel based on both the source and destination MAC addresses. This forwarding method, a combination source-MAC and destination-MAC address forwarding methods of load distribution, can be used if it is not clear whether source-MAC or destination-MAC address forwarding is better suited on a particular device. With source-and-destination MAC-address forwarding, packets sent from host A to host B, host A to host C, and host C to host B could all use different ports in the channel.

IP Address Forwarding

With source-IP address-based forwarding, packets are distributed across the ports in the EtherChannel based on the source-IP address of the incoming packet. To provide load balancing, packets from different IP addresses use different ports in the channel, and packets from the same IP address use the same port in the channel.

With destination-IP address-based forwarding, packets are distributed across the ports in the EtherChannel based on the destination-IP address of the incoming packet. To provide load balancing, packets from the same IP source address sent to different IP destination addresses could be sent on different ports in the channel. Packets sent from different source IP addresses to the same destination IP address are always sent on the same port in the channel.

With source-and-destination IP address-based forwarding, packets are distributed across the ports in the EtherChannel based on both the source and destination IP addresses of the incoming packet. This forwarding method, a combination of source-IP and destination-IP address-based forwarding, can be used if it is not clear whether source-IP or destination-IP address-based forwarding is better suited on a particular device. In this method, packets sent from the IP address A to IP address B, from IP address A to IP address C, and from IP address C to IP address B could all use different ports in the channel.

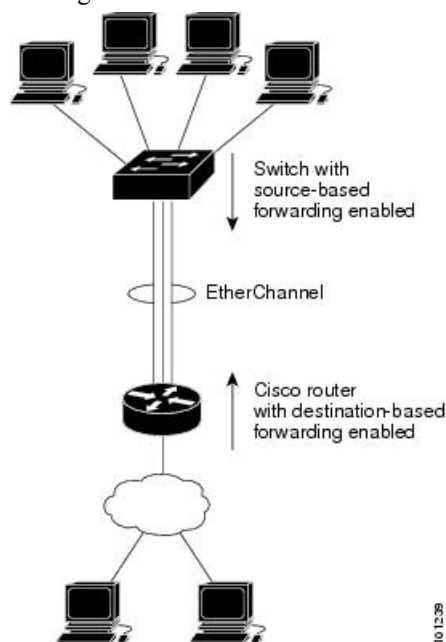
Load-Balancing Advantages

Different load-balancing methods have different advantages, and the choice of a particular load-balancing method should be based on the position of the device in the network and the kind of traffic that needs to be load-distributed.

Figure 50: Load Distribution and Forwarding Methods

In the following figure, an EtherChannel of four workstations communicates with a router. Because the router is a single MAC-address device, source-based forwarding on the device EtherChannel ensures that the device uses all available bandwidth to the router. The router is configured for destination-based forwarding because

the large number of workstations ensures that the traffic is evenly distributed from the router EtherChannel.



Use the option that provides the greatest variety in your configuration. For example, if the traffic on a channel is going only to a single MAC address, using the destination-MAC address always chooses the same link in the channel. Using source addresses or IP addresses might result in better load-balancing.

EtherChannel Load Deferral Overview

In an Instant Access system, the EtherChannel Load Deferral feature allows ports to be bundled into port channels, but prevents the assignment of group mask values to these ports. This prevents the traffic from being forwarded to new instant access stack members and reduce data loss following a stateful switchover (SSO).

Cisco Catalyst Instant Access creates a single network touch point and a single point of configuration across distribution and access layer switches. Instant Access enables the merging of physical distribution and access layer switches into a single logical entity with a single point of configuration, management, and troubleshooting. The following illustration represents a sample network where an Instant Access system interacts with a switch (Catalyst 2960-X Series Switches) that is connected via a port channel to stacked clients (Member 1 and Member 2).

When the EtherChannel Load Deferral feature is configured and a new Instant Access client stack member comes up, ports of this newly-joined stack member is bundled into the port channel. In the transition period, the data path is not fully established on the distribution switch (Catalyst 6000 Series Switches), and traffic originating from the access layer switch (Catalyst 2960-X Series Switches) reaches the non-established ports and the traffic gets lost.

When load share deferral is enabled on a port channel, the assignment of a member port's load share is delayed for a period that is configured globally by the **port-channel load-defer** command. During the deferral period, the load share of a deferred member port is set to 0. In this state, the deferred port is capable of receiving data and control traffic, and of sending control traffic, but the port is prevented from sending data traffic to the virtual switching system (VSS). Upon expiration of the global deferral timer, the deferred member port exits the deferral state and the port assumes its normal configured load share.

Load share deferral is applied only if at least one member port of the port channel is currently active with a nonzero load share. If a port enabled for load share deferral is the first member bringing up the EtherChannel, the deferral feature does not apply and the port will forward traffic immediately.

This feature is enabled on a per port-channel basis; however, the load deferral timer is configured globally and not per port-channel. As a result, when a new port is bundled, the timer starts only if it is not already running. If some other ports are already deferred then the new port will be deferred only for the remaining amount of time.

The load deferral is stopped as soon as a member in one of the deferred port channels is unbundled. As a result, all the ports that were deferred is assigned a group-mask in the event of an unbundling during the deferral period.



Note When you try to enable this feature on a stack member switch, the following message is displayed:

```
Load share deferral is supported only on stand-alone stack.
```

EtherChannel and Device Stacks

If a stack member that has ports participating in an EtherChannel fails or leaves the stack, the active device removes the failed stack member device ports from the EtherChannel. The remaining ports of the EtherChannel, if any, continue to provide connectivity.

When a device is added to an existing stack, the new device receives the running configuration from the active device and updates itself with the EtherChannel-related stack configuration. The stack member also receives the operational information (the list of ports that are up and are members of a channel).

When two stacks merge that have EtherChannels configured between them, self-looped ports result. Spanning tree detects this condition and acts accordingly. Any PAgP or LACP configuration on a winning device stack is not affected, but the PAgP or LACP configuration on the losing device stack is lost after the stack reboots.

For a mixed stack containing one or more Catalyst 2960-S switches, we recommend that you configure no more than six EtherChannels on the stack.

Device Stack and PAgP

With PAgP, if the active device fails or leaves the stack, the standby device becomes the new active device. The new active device synchronizes the configuration of the stack members to that of the active device. The PAgP configuration is not affected after an active device change unless the EtherChannel has ports residing on the old active device.

Switch Stacks and LACP

With LACP, the system ID uses the stack MAC address from the active switch, and if the active switch changes, the LACP system ID can change. If the LACP system ID changes, the entire EtherChannel will flap, and there will be an STP reconvergence. Use the **stack-mac persistent timer** command to control whether or not the stack MAC address changes during a active switch failover.

Default EtherChannel Configuration

The default EtherChannel configuration is described in this table.

Table 55: Default EtherChannel Configuration

Feature	Default Setting
Channel groups	None assigned.
Port-channel logical interface	None defined.
PAgP mode	No default.
PAgP learn method	Aggregate-port learning on all ports.
PAgP priority	128 on all ports.
LACP mode	No default.
LACP learn method	Aggregate-port learning on all ports.
LACP port priority	32768 on all ports.
LACP system priority	32768.
LACP system ID	LACP system priority and the device or stack MAC address.
Load-balancing	Load distribution on the device is based on the source-MAC address of the incoming packet.

EtherChannel Configuration Guidelines

If improperly configured, some EtherChannel ports are automatically disabled to avoid network loops and other problems. Follow these guidelines to avoid configuration problems:

- Do not try to configure more than 24 EtherChannels on the device or device stack.
- In a mixed switch stack that contains one or more Catalyst 2960-S switches, do not configure more than six EtherChannels on the switch stack.
- Configure a PAgP EtherChannel with up to eight Ethernet ports of the same type.
- Configure a LACP EtherChannel with up to 16 Ethernet ports of the same type. Up to eight ports can be active, and up to eight ports can be in standby mode.
- Configure all ports in an EtherChannel to operate at the same speeds and duplex modes.
- Enable all ports in an EtherChannel. A port in an EtherChannel that is disabled by using the **shutdown** interface configuration command is treated as a link failure, and its traffic is transferred to one of the remaining ports in the EtherChannel.

- When a group is first created, all ports follow the parameters set for the first port to be added to the group. If you change the configuration of one of these parameters, you must also make the changes to all ports in the group:
 - Allowed-VLAN list
 - Spanning-tree path cost for each VLAN
 - Spanning-tree port priority for each VLAN
 - Spanning-tree Port Fast setting
- Do not configure a port to be a member of more than one EtherChannel group.
- Do not configure an EtherChannel in both the PAgP and LACP modes. EtherChannel groups running PAgP and LACP can coexist on the same device or on different devices in the stack. Individual EtherChannel groups can run either PAgP or LACP, but they cannot interoperate.
- Do not configure a secure port as part of an EtherChannel or the reverse.
- Do not configure a port that is an active or a not-yet-active member of an EtherChannel as an IEEE 802.1x port. If you try to enable IEEE 802.1x on an EtherChannel port, an error message appears, and IEEE 802.1x is not enabled.
- If EtherChannels are configured on device interfaces, remove the EtherChannel configuration from the interfaces before globally enabling IEEE 802.1x on a device by using the **dot1x system-auth-control** global configuration command.
- For cross-stack EtherChannel configurations, ensure that all ports targeted for the EtherChannel are either configured for LACP or are manually configured to be in the channel group using the **channel-group channel-group-number mode on** interface configuration command. The PAgP protocol is not supported on cross- stack EtherChannels.

Layer 2 EtherChannel Configuration Guidelines

When configuring Layer 2 EtherChannels, follow these guidelines:

- Assign all ports in the EtherChannel to the same VLAN, or configure them as trunks. Ports with different native VLANs cannot form an EtherChannel.
- An EtherChannel supports the same allowed range of VLANs on all the ports in a trunking Layer 2 EtherChannel. If the allowed range of VLANs is not the same, the ports do not form an EtherChannel even when PAgP is set to the **auto** or **desirable** mode.
- Ports with different spanning-tree path costs can form an EtherChannel if they are otherwise compatibly configured. Setting different spanning-tree path costs does not, by itself, make ports incompatible for the formation of an EtherChannel.

Layer 3 EtherChannel Configuration Guidelines

- For Layer 3 EtherChannels, assign the Layer 3 address to the port-channel logical interface, not to the physical ports in the channel.

Auto-LAG

The auto-LAG feature provides the ability to auto create EtherChannels on ports connected to a switch. By default, auto-LAG is disabled globally and is enabled on all port interfaces. The auto-LAG applies to a switch only when it is enabled globally.

On enabling auto-LAG globally, the following scenarios are possible:

- All port interfaces participate in creation of auto EtherChannels provided the partner port interfaces have EtherChannel configured on them. For more information, see the *"The supported auto-LAG configurations between the actor and partner devices"* table below.
- Ports that are already part of manual EtherChannels cannot participate in creation of auto EtherChannels.
- When auto-LAG is disabled on a port interface that is already a part of an auto created EtherChannel, the port interface will unbundle from the auto EtherChannel.

The following table shows the supported auto-LAG configurations between the actor and partner devices:

Table 56: The supported auto-LAG configurations between the actor and partner devices

Actor/Partner	Active	Passive	Auto
Active	Yes	Yes	Yes
Passive	Yes	No	Yes
Auto	Yes	Yes	Yes

On disabling auto-LAG globally, all auto created Etherchannels become manual EtherChannels.

You cannot add any configurations in an existing auto created EtherChannel. To add, you should first convert it into a manual EtherChannel by executing the **port-channel<channel-number>persistent**.



Note Auto-LAG uses the LACP protocol to create auto EtherChannel. Only one EtherChannel can be automatically created with the unique partner devices.

Auto-LAG Configuration Guidelines

Follow these guidelines when configuring the auto-LAG feature.

- When auto-LAG is enabled globally and on the port interface , and if you do not want the port interface to become a member of the auto EtherChannel, disable the auto-LAG on the port interface.
- A port interface will not bundle to an auto EtherChannel when it is already a member of a manual EtherChannel. To allow it to bundle with the auto EtherChannel, first unbundle the manual EtherChannel on the port interface.
- When auto-LAG is enabled and auto EtherChannel is created, you can create multiple EtherChannels manually with the same partner device. But by default, the port tries to create auto EtherChannel with the partner device.
- The auto-LAG is supported only on Layer 2 EtherChannel. It is not supported on Layer 3 interface and Layer 3 EtherChannel.

- The auto-LAG is supported on cross-stack EtherChannel.

How to Configure EtherChannels

After you configure an EtherChannel, configuration changes applied to the port-channel interface apply to all the physical ports assigned to the port-channel interface, and configuration changes applied to the physical port affect only the port where you apply the configuration.

Configuring Layer 2 EtherChannels

You configure Layer 2 EtherChannels by assigning ports to a channel group with the **channel-group** interface configuration command. This command automatically creates the port-channel logical interface.

If you enabled PAgP on a port in the **auto** or **desirable** mode, you must reconfigure it for either the **on** mode or the LACP mode before adding this port to a cross-stack EtherChannel. PAgP does not support cross-stack EtherChannels.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **switchport mode** {access | trunk}
4. **switchport access vlan** *vlan-id*
5. **channel-group** *channel-group-number* **mode** {auto [non-silent] | desirable [non-silent] | on } | { active | passive }
6. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 2	interface <i>interface-id</i> Example: Device(config)# interface gigabitethernet1/0/1	Specifies a physical port, and enters interface configuration mode. Valid interfaces are physical ports. For a PAgP EtherChannel, you can configure up to eight ports of the same type and speed for the same group. For a LACP EtherChannel, you can configure up to 16 Ethernet ports of the same type. Up to eight ports can be active, and up to eight ports can be in standby mode.

	Command or Action	Purpose
Step 3	<p>switchport mode {access trunk}</p> <p>Example:</p> <pre>Device(config-if)# switchport mode access</pre>	<p>Assigns all ports as static-access ports in the same VLAN, or configure them as trunks.</p> <p>If you configure the port as a static-access port, assign it to only one VLAN. The range is 1 to 4094.</p>
Step 4	<p>switchport access vlan <i>vlan-id</i></p> <p>Example:</p> <pre>Device(config-if)# switchport access vlan 22</pre>	<p>(Optional) If you configure the port as a static-access port, assign it to only one VLAN. The range is 1 to 4094.</p>
Step 5	<p>channel-group <i>channel-group-number</i> mode {auto [non-silent] desirable [non-silent] on } { active passive }</p> <p>Example:</p> <pre>Device(config-if)# channel-group 5 mode auto</pre>	<p>Assigns the port to a channel group, and specifies the PAgP or the LACP mode.</p> <p>For <i>channel-group-number</i>, the range is 1 to 48.</p> <p>For mode, select one of these keywords:</p> <ul style="list-style-type: none"> • auto —Enables PAgP only if a PAgP device is detected. It places the port into a passive negotiating state, in which the port responds to PAgP packets it receives but does not start PAgP packet negotiation. This keyword is not supported when EtherChannel members are from different devices in the device stack. • desirable —Unconditionally enables PAgP. It places the port into an active negotiating state, in which the port starts negotiations with other ports by sending PAgP packets. This keyword is not supported when EtherChannel members are from different devices in the device stack. • on —Forces the port to channel without PAgP or LACP. In the on mode, an EtherChannel exists only when a port group in the on mode is connected to another port group in the on mode. • non-silent —(Optional) If your device is connected to a partner that is PAgP-capable, configures the device port for nonsilent operation when the port is in the auto or desirable mode. If you do not specify non-silent, silent is assumed. The silent setting is for connections to file servers or packet analyzers. This setting allows PAgP to operate, to attach the port to a channel group, and to use the port for transmission. • active —Enables LACP only if a LACP device is detected. It places the port into an active negotiating state in which the port starts negotiations with other ports by sending LACP packets.

	Command or Action	Purpose
		<ul style="list-style-type: none"> • passive –Enables LACP on the port and places it into a passive negotiating state in which the port responds to LACP packets that it receives, but does not start LACP packet negotiation.
Step 6	end Example: Device(config-if) # end	Returns to privileged EXEC mode.

Configuring Layer 3 EtherChannels

To configure Layer 3 EtherChannels, you create the port-channel logical interface and then put the Ethernet ports into the port channel as described in the next two sections.

Creating Port-Channel Logical Interfaces

When configuring Layer 3 EtherChannels, you should first manually create the port-channel logical interface by using the **interface port-channel** global configuration command. Then put the logical interface into the channel group by using the **channel-group** interface configuration command.



Note To move an IP address from a physical port to an EtherChannel, you must delete the IP address from the physical port before configuring it on the port-channel interface.

Follow these steps to create a port-channel interface for a Layer 3 EtherChannel. This procedure is required.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface port-channel** *port-channel-number*
4. **no switchport**
5. **ip address** *ip-address mask*
6. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.

	Command or Action	Purpose
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	interface port-channel <i>port-channel-number</i> Example: Device(config)# interface port-channel 5	Specifies the port-channel logical interface, and enters interface configuration mode. For <i>port-channel-number</i> , the range is 1 to 48.
Step 4	no switchport Example: Device(config-if)# no switchport	Puts the interface into Layer 3 mode.
Step 5	ip address <i>ip-address mask</i> Example: Device(config-if)# ip address ip address 172.10.20.10 255.255.255.0	Assigns an IP address and subnet mask to the EtherChannel.
Step 6	end Example: Device(config-if)# end	Returns to privileged EXEC mode.

This example shows how to create the logical port channel 5 and assign 172.10.20.10 as its IP address:

```
Switch# configure terminal
Switch(config)# interface port-channel 5
Switch(config-if)# no switchport
Switch(config-if)# ip address 172.10.20.10 255.255.255.0
Switch(config-if)# end
```

Configuring the Physical Interfaces

Follow these steps to assign an Ethernet port to a Layer 3 EtherChannel. This procedure is required.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **no ip address**
5. **no switchport**
6. **channel-group** *channel-group-number mode* { **auto** [**non-silent**] | **desirable** [**non-silent**] | **on** } | { **active** | **passive** }

7. end

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	interface <i>interface-id</i> Example: Device(config)# interface gigabitethernet 1/0/2	Specifies a physical port, and enters interface configuration mode. Valid interfaces include physical ports. For a PAgP EtherChannel, you can configure up to eight ports of the same type and speed for the same group. For a LACP EtherChannel, you can configure up to 16 Ethernet ports of the same type. Up to eight ports can be active, and up to eight ports can be in standby mode.
Step 4	no ip address Example: Device(config-if)# no ip address	Ensures that there is no IP address assigned to the physical port.
Step 5	no switchport Example: Device(config-if)# no switchport	Puts the port into Layer 3 mode.
Step 6	channel-group <i>channel-group-number</i> mode { auto [non-silent] desirable [non-silent] on } { active passive } Example: Device(config-if)# channel-group 5 mode auto	Assigns the port to a channel group, and specifies the PAgP or the LACP mode. For <i>channel-group-number</i> , the range is 1 to 48. This number must be that of a previously created port channel (logical port). For mode , select one of these keywords: <ul style="list-style-type: none"> • auto—Enables PAgP only if a PAgP device is detected. It places the port into a passive negotiating state, in which the port responds to PAgP packets it receives but does not start PAgP packet negotiation. This keyword is not supported when EtherChannel members are from different devices in the device stack.

	Command or Action	Purpose
		<ul style="list-style-type: none"> • desirable—Unconditionally enables PAgP. It places the port into an active negotiating state, in which the port starts negotiations with other ports by sending PAgP packets. This keyword is not supported when EtherChannel members are from different devices in the device stack. • on—Forces the port to channel without PAgP or LACP. In the on mode, an EtherChannel exists only when a port group in the on mode is connected to another port group in the on mode. • non-silent—(Optional) If your device is connected to a partner that is PAgP capable, configures the device port for nonsilent operation when the port is in the auto or desirable mode. If you do not specify non-silent, silent is assumed. The silent setting is for connections to file servers or packet analyzers. This setting allows PAgP to operate, to attach the port to a channel group, and to use the port for transmission. • active—Enables LACP only if a LACP device is detected. It places the port into an active negotiating state in which the port starts negotiations with other ports by sending LACP packets. • passive—Enables LACP on the port and places it into a passive negotiating state in which the port responds to LACP packets that it receives, but does not start LACP packet negotiation.
Step 7	end Example: Device(config-if) # end	Returns to privileged EXEC mode.

Configuring EtherChannel Load-Balancing

You can configure EtherChannel load-balancing by using source-based or destination-based forwarding methods.

This task is optional.

SUMMARY STEPS

1. **configure terminal**
2. **port-channel load-balance** { **dst-ip** | **dst-mac** | **src-dst-ip** | **src-dst-mac** | **src-ip** | **src-mac** }
3. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>Device# configure terminal</pre>	Enters global configuration mode.
Step 2	port-channel load-balance { dst-ip dst-mac src-dst-ip src-dst-mac src-ip src-mac } Example: <pre>Device(config)# port-channel load-balance src-mac</pre>	Configures an EtherChannel load-balancing method. The default is src-mac . Select one of these load-distribution methods: <ul style="list-style-type: none"> • dst-ip—Specifies destination-host IP address. • dst-mac—Specifies the destination-host MAC address of the incoming packet. • src-dst-ip—Specifies the source and destination host IP address. • src-dst-mac—Specifies the source and destination host MAC address. • src-ip—Specifies the source host IP address. • src-mac—Specifies the source MAC address of the incoming packet.
Step 3	end Example: <pre>Device(config)# end</pre>	Returns to privileged EXEC mode.

Configuring Port Channel Load Deferral

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **port-channel load-defer** *seconds*
4. **interface** *type number*
5. **port-channel load-defer**
6. **end**
7. **show etherchannel** *channel-group* **port-channel**
8. **show platform pm group-masks**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. <ul style="list-style-type: none">• Enter your password if prompted.
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	port-channel load-defer <i>seconds</i> Example: Switch(config)# port-channel load-defer 60	Configures the port load share deferral interval for all port channels. <ul style="list-style-type: none">• <i>seconds</i>—The time interval during which load sharing is initially 0 for deferred port channels. The range is 1 to 1800 seconds; the default is 120 seconds
Step 4	interface <i>type number</i> Example: Switch(config)# interface port-channel 10	Configures a port channel interface and enters interface configuration mode.
Step 5	port-channel load-defer Example: Switch(config-if)# port-channel load-defer	Enables port load share deferral on the port channel.
Step 6	end Example: Switch(config-if)# end	Exits interface configuration mode and returns to privileged EXEC mode.
Step 7	show etherchannel <i>channel-group</i> port-channel Example: Switch# show etherchannel 1 port-channel	Displays port channel information.
Step 8	show platform pm group-masks Example: Switch# show platform pm group-masks	Display EtherChannel group masks information.

Example

The following is sample output from the **show etherchannel *channel-group* port-channel** command. If the *channel-group* argument is not specified, the command displays information about all channel groups are displayed.

```
Switch# show etherchannel 1 port-channel
```

```
Port-channels in the group:
-----
```

```

Port-channel: Po1
-----

Age of the Port-channel   = 0d:00h:37m:08s
Logical slot/port        = 9/1           Number of ports = 0
GC                       = 0x00000000    HotStandBy port = null
Port state               = Port-channel Ag-Not-Inuse
Protocol                 = -
Port security            = Disabled
Load share deferral     = Enabled      defer period = 120 sec   time left = 0 sec

```

The following is sample output from the **show platform pm group-masks** command. Deferred ports have the group mask of 0xFFFF, when the defer timer is running.

```

Switch# show platform pm group-masks

=====
                        Etherchannel members and group masks table
Group #ports group frame-dist slot port mask interface index
-----
  1   0       1   src-mac
  2   0       2   src-mac
  3   0       3   src-mac
  4   0       4   src-mac
  5   0       5   src-mac
  6   0       6   src-mac
  7   0       7   src-mac
  8   0       8   src-mac
  9   0       9   src-mac
 10   3      10   src-mac
                        1   12   0000 Gil/0/12  3
                        1   10   FFFF Gil/0/10  6
                        1   11   FFFF Gil/0/11  7
 11   0      11   src-mac
 12   0      12   src-mac
 13   0      13   src-mac
 14   0      14   src-mac
 15   0      15   src-mac

```

Configuring the PAgP Learn Method and Priority

This task is optional.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **pagp learn-method physical-port**
4. **pagp port-priority** *priority*
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 2	interface interface-id Example: Device(config)# interface gigabitethernet 1/0/2	Specifies the port for transmission, and enters interface configuration mode.
Step 3	pagp learn-method physical-port Example: Device(config-if)# pagp learn-method physical port	<p>Selects the PAgP learning method.</p> <p>By default, aggregation-port learning is selected, which means the device sends packets to the source by using any of the ports in the EtherChannel. With aggregate-port learning, it is not important on which physical port the packet arrives.</p> <p>Selects physical-port to connect with another device that is a physical learner.</p> <p>Make sure to configure the port-channel load-balance global configuration command to src-mac.</p> <p>The learning method must be configured the same at both ends of the link.</p>
Step 4	pagp port-priority priority Example: Device(config-if)# pagp port-priority 200	<p>Assigns a priority so that the selected port is chosen for packet transmission.</p> <p>For <i>priority</i>, the range is 0 to 255. The default is 128. The higher the priority, the more likely that the port will be used for PAgP transmission.</p>
Step 5	end Example: Device(config-if)# end	Returns to privileged EXEC mode.

Configuring LACP Hot-Standby Ports

When enabled, LACP tries to configure the maximum number of LACP-compatible ports in a channel, up to a maximum of 16 ports. Only eight LACP links can be active at one time. The software places any additional links in a hot-standby mode. If one of the active links becomes inactive, a link that is in the hot-standby mode becomes active in its place.

If you configure more than eight links for an EtherChannel group, the software automatically decides which of the hot-standby ports to make active based on the LACP priority. To every link between systems that operate LACP, the software assigns a unique priority made up of these elements (in priority order):

- LACP system priority
- System ID (the device MAC address)
- LACP port priority
- Port number

In priority comparisons, numerically lower values have higher priority. The priority decides which ports should be put in standby mode when there is a hardware limitation that prevents all compatible ports from aggregating.

Determining which ports are active and which are hot standby is a two-step procedure. First the system with a numerically lower system priority and system ID is placed in charge of the decision. Next, that system decides which ports are active and which are hot standby, based on its values for port priority and port number. The port priority and port number values for the other system are not used.

You can change the default values of the LACP system priority and the LACP port priority to affect how the software selects active and standby links.

Configuring the LACP System Priority

You can configure the system priority for all the EtherChannels that are enabled for LACP by using the **lacp system-priority** global configuration command. You cannot configure a system priority for each LACP-configured channel. By changing this value from the default, you can affect how the software selects active and standby links.

You can use the **show etherchannel summary** privileged EXEC command to see which ports are in the hot-standby mode (denoted with an H port-state flag).

Follow these steps to configure the LACP system priority. This procedure is optional.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **lacp system-priority** *priority*
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example:	Enters global configuration mode.

	Command or Action	Purpose
	Device# <code>configure terminal</code>	
Step 3	lACP system-priority <i>priority</i> Example: Device(config)# <code>lACP system-priority 32000</code>	Configures the LACP system priority. The range is 1 to 65535. The default is 32768. The lower the value, the higher the system priority.
Step 4	end Example: Device(config)# <code>end</code>	Returns to privileged EXEC mode.

Configuring the LACP Port Priority

By default, all ports use the same port priority. If the local system has a lower value for the system priority and the system ID than the remote system, you can affect which of the hot-standby links become active first by changing the port priority of LACP EtherChannel ports to a lower value than the default. The hot-standby ports that have lower port numbers become active in the channel first. You can use the **show etherchannel summary** privileged EXEC command to see which ports are in the hot-standby mode (denoted with an H port-state flag).



Note If LACP is not able to aggregate all the ports that are compatible (for example, the remote system might have more restrictive hardware limitations), all the ports that cannot be actively included in the EtherChannel are put in the hot-standby state and are used only if one of the channeled ports fails.

Follow these steps to configure the LACP port priority. This procedure is optional.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **lACP port-priority** *priority*
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> <code>enable</code>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.

	Command or Action	Purpose
Step 2	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode.
Step 3	interface <i>interface-id</i> Example: Device(config)# <code>interface gigabitethernet 1/0/2</code>	Specifies the port to be configured, and enters interface configuration mode.
Step 4	lacp port-priority <i>priority</i> Example: Device(config-if)# <code>lacp port-priority 32000</code>	Configures the LACP port priority. The range is 1 to 65535. The default is 32768. The lower the value, the more likely that the port will be used for LACP transmission.
Step 5	end Example: Device(config-if)# <code>end</code>	Returns to privileged EXEC mode.

Configuring the LACP Port Channel Min-Links Feature

You can specify the minimum number of active ports that must be in the link-up state and bundled in an EtherChannel for the port channel interface to transition to the link-up state. Using EtherChannel min-links, you can prevent low-bandwidth LACP EtherChannels from becoming active. Port channel min-links also cause LACP EtherChannels to become inactive if they have too few active member ports to supply the required minimum bandwidth.

To configure the minimum number of links that are required for a port channel. Perform the following tasks.

SUMMARY STEPS

1. `enable`
2. `configure terminal`
3. `interface port-channel channel-number`
4. `port-channel min-links min-links-number`
5. `end`

DETAILED STEPS

	Command or Action	Purpose
Step 1	<code>enable</code>	Enables privileged EXEC mode.

	Command or Action	Purpose
	Example: Device> enable	<ul style="list-style-type: none"> Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	interface port-channel <i>channel-number</i> Example: Device(config)# interface port-channel 2	Enters interface configuration mode for a port-channel. For <i>channel-number</i> , the range is 1 to 63.
Step 4	port-channel min-links <i>min-links-number</i> Example: Device(config-if)# port-channel min-links 3	Specifies the minimum number of member ports that must be in the link-up state and bundled in the EtherChannel for the port channel interface to transition to the link-up state. For <i>min-links-number</i> , the range is 2 to 8.
Step 5	end Example: Device(config)# end	Returns to privileged EXEC mode.

Configuring LACP Fast Rate Timer

You can change the LACP timer rate to modify the duration of the LACP timeout. Use the **lACP rate** command to set the rate at which LACP control packets are received by an LACP-supported interface. You can change the timeout rate from the default rate (30 seconds) to the fast rate (1 second). This command is supported only on LACP-enabled interfaces.

SUMMARY STEPS

- enable**
- configure terminal**
- interface {fastethernet | gigabitethernet | tengigabitethernet} slot/port**
- lACP rate {normal | fast}**
- end**
- show lACP internal**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	interface {fastethernet gigabitethernet tengigabitethernet} slot/port Example: Device(config)# interface gigabitEthernet 2/1	Configures an interface and enters interface configuration mode.
Step 4	lACP rate {normal fast} Example: Device(config-if)# lACP rate fast	Configures the rate at which LACP control packets are received by an LACP-supported interface. <ul style="list-style-type: none"> • To reset the timeout rate to its default, use the no lACP rate command.
Step 5	end Example: Device(config)# end	Returns to privileged EXEC mode.
Step 6	show lACP internal Example: Device# show lACP internal Device# show lACP counters	Verifies your configuration.

Configuring Auto-LAG Globally

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **[no] port-channel auto**
4. **end**
5. **show etherchannel auto**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> <code>enable</code>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode.
Step 3	[no] port-channel auto Example: Device(config)# <code>port-channel auto</code>	Enables the auto-LAG feature on a switch globally. Use the no form of this command to disable the auto-LAG feature on the switch globally. Note By default, the auto-LAG feature is enabled on the port.
Step 4	end Example: Device(config)# <code>end</code>	Returns to privileged EXEC mode.
Step 5	show etherchannel auto Example: Device# <code>show etherchannel auto</code>	Displays that EtherChannel is created automatically.

Configuring Auto-LAG on a Port Interface

SUMMARY STEPS

1. `enable`
2. `configure terminal`
3. `interface interface-id`
4. `[no] channel-group auto`
5. `end`
6. `show etherchannel auto`

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example:	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.

	Command or Action	Purpose
	Device> <code>enable</code>	
Step 2	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode.
Step 3	interface <i>interface-id</i> Example: Device(config)# <code>interface gigabitethernet 1/0/1</code>	Specifies the port interface to be enabled for auto-LAG, and enters interface configuration mode.
Step 4	[no] channel-group auto Example: Device(config-if)# <code>channel-group auto</code>	(Optional) Enables auto-LAG feature on individual port interface. Use the no form of this command to disable the auto-LAG feature on individual port interface. Note By default, the auto-LAG feature is enabled on the port.
Step 5	end Example: Device(config-if)# <code>end</code>	Returns to privileged EXEC mode.
Step 6	show etherchannel auto Example: Device# <code>show etherchannel auto</code>	Displays that EtherChannel is created automatically.

What to do next

Configuring Persistence with Auto-LAG

You use the persistence command to convert the auto created EtherChannel into a manual one and allow you to add configuration on the existing EtherChannel.

SUMMARY STEPS

1. `enable`
2. `port-channel channel-number persistent`
3. `show etherchannel summary`

DETAILED STEPS

	Command or Action	Purpose
Step 1	<code>enable</code>	Enables privileged EXEC mode.

	Command or Action	Purpose
	Example: Device> enable	<ul style="list-style-type: none"> Enter your password if prompted.
Step 2	port-channel <i>channel-number</i> persistent Example: Device# port-channel 1 persistent	Converts the auto created EtherChannel into a manual one and allows you to add configuration on the EtherChannel.
Step 3	show etherchannel summary Example: Device# show etherchannel summary	Displays the EtherChannel information.

Monitoring EtherChannel, PAgP, and LACP Status

You can display EtherChannel, PAgP, and LACP status using the commands listed in this table.

Table 57: Commands for Monitoring EtherChannel, PAgP, and LACP Status

Command	Description
clear lacp { <i>channel-group-number</i> counters counters }	Clears LACP channel-group information and traffic counters.
clear pagp { <i>channel-group-number</i> counters counters }	Clears PAgP channel-group information and traffic counters.
show etherchannel [<i>channel-group-number</i> { detail load-balance port port-channel protocol summary }] [detail load-balance port port-channel protocol auto summary]	Displays EtherChannel information in a brief, detailed, and one-line summary form. Also displays the load-balance or frame-distribution scheme, port, port-channel, protocol, and Auto-LAG information.
show pagp [<i>channel-group-number</i>] { counters internal neighbor }	Displays PAgP information such as traffic information, the internal PAgP configuration, and neighbor information.
show pagp [<i>channel-group-number</i>] dual-active	Displays the dual-active detection status.
show lacp [<i>channel-group-number</i>] { counters internal neighbor sys-id }	Displays LACP information such as traffic information, the internal LACP configuration, and neighbor information.
show running-config	Verifies your configuration entries.
show etherchannel load-balance	Displays the load balance or frame distribution scheme among ports in the port channel.

Configuration Examples for Configuring EtherChannels

Configuring Layer 2 EtherChannels: Examples

This example shows how to configure an EtherChannel on a single device in the stack. It assigns two ports as static-access ports in VLAN 10 to channel 5 with the PAgP mode **desirable**:

```
Device# configure terminal
Device(config)# interface range gigabitethernet2/0/1 -2
Device(config-if-range)# switchport mode access
Device(config-if-range)# switchport access vlan 10
Device(config-if-range)# channel-group 5 mode desirable non-silent
Device(config-if-range)# end
```

This example shows how to configure an EtherChannel on a single device in the stack. It assigns two ports as static-access ports in VLAN 10 to channel 5 with the LACP mode **active** :

```
Device# configure terminal
Device(config)# interface range gigabitethernet2/0/1 -2
Device(config-if-range)# switchport mode access
Device(config-if-range)# switchport access vlan 10
Device(config-if-range)# channel-group 5 mode active
Device(config-if-range)# end
```

This example shows how to configure a cross-stack EtherChannel. It uses LACP passive mode and assigns two ports on stack member 1 and one port on stack member 2 as static-access ports in VLAN 10 to channel 5:

```
Device# configure terminal
Device(config)# interface range gigabitethernet2/0/4 -5
Device(config-if-range)# switchport mode access
Device(config-if-range)# switchport access vlan 10
Device(config-if-range)# channel-group 5 mode passive
Device(config-if-range)# exit
Device(config)# interface gigabitethernet3/0/3
Device(config-if)# switchport mode access
Device(config-if)# switchport access vlan 10
Device(config-if)# channel-group 5 mode passive
Device(config-if)# exit
```

PoE or LACP negotiation errors may occur if you configure two ports from switch to the access point (AP). This scenario can be avoided if the port channel configuration is on the switch side. For more details, see the following example:

```
interface Port-channell
  switchport access vlan 20
  switchport mode access
  switchport nonegotiate
  no port-channel standalone-disable <--this one
  spanning-tree portfast
```



Note If the port reports LACP errors on port flap, you should include the following command as well: **no errdisable detect cause pagp-flap**

Configuring Port-Channel Logical Interfaces: Example

This example shows how to create the logical port channel 5 and assign 172.10.20.10 as its IP address:

```
Device# configure terminal
Device(config)# interface port-channel 5
Device(config-if)# no switchport
Device(config-if)# ip address 172.10.20.10 255.255.255.0
Device(config-if)# end
```

Configuring EtherChannel Physical Interfaces: Examples

This example shows how to configure an EtherChannel. It assigns two ports to channel 5 with the LACP mode active:

```
Device# configure terminal
Device(config)# interface range gigabitethernet2/0/1 -2
Device(config-if-range)# no ip address
Device(config-if-range)# no switchport
Device(config-if-range)# channel-group 5 mode active
Device(config-if-range)# end
```

This example shows how to configure a cross-stack EtherChannel. It assigns two ports on stack member 2 and one port on stack member 3 to channel 7 using LACP active mode:

```
Device# configure terminal
Device(config)# interface range gigabitethernet2/0/4 -5
Device(config-if-range)# no ip address
Device(config-if-range)# no switchport
Device(config-if-range)# channel-group 7 mode active
Device(config-if-range)# exit
Device(config)# interface gigabitethernet3/0/3
Device(config-if)# no ip address
Device(config-if)# no switchport
Device(config-if)# channel-group 7 mode active
Device(config-if)# exit
```


Example: Configuring Port Channel Load Deferral

```
Switch# configure terminal
Switch(config)# port-channel load-defer 60
Switch(config)# interface port-channel 10
Switch(config-if)# port-channel load-defer
Switch(config-if)# end
```

Configuring Auto LAG: Examples

This example shows how to configure Auto-LAG on a switch

```
device> enable
device# configure terminal
device (config)# port-channel auto
device (config-if)# end
device# show etherchannel auto
```

The following example shows the summary of EtherChannel that was created automatically.

```
device# show etherchannel auto
Flags: D - down          P - bundled in port-channel
       I - stand-alone  s - suspended
       H - Hot-standby (LACP only)
       R - Layer3       S - Layer2
       U - in use       f - failed to allocate aggregator
       M - not in use, minimum links not met
       u - unsuitable for bundling
       w - waiting to be aggregated
       d - default port
       A - formed by Auto LAG
```

```
Number of channel-groups in use: 1
Number of aggregators:          1
```

Group	Port-channel	Protocol	Ports
1	Pol(SUA)	LACP	Gi1/0/45(P) Gi2/0/21(P) Gi3/0/21(P)

The following example shows the summary of auto EtherChannel after executing the **port-channel 1 persistent** command.

```
device# port-channel 1 persistent

device# show etherchannel summary
Switch# show etherchannel summary
Flags: D - down          P - bundled in port-channel
       I - stand-alone  s - suspended
       H - Hot-standby (LACP only)
       R - Layer3       S - Layer2
       U - in use       f - failed to allocate aggregator
       M - not in use, minimum links not met
       u - unsuitable for bundling
       w - waiting to be aggregated
       d - default port
       A - formed by Auto LAG
```

```
Number of channel-groups in use: 1
Number of aggregators:          1
```

```

Group  Port-channel  Protocol  Ports
-----+-----+-----+-----
1      Po1(SU)          LACP      Gi1/0/45(P) Gi2/0/21(P) Gi3/0/21(P)

```

Configuring LACP Port Channel Min-Links: Examples

This example shows how to configure LACP port-channel min-links:

```

device > enable
device# configure terminal
device(config)# interface port-channel 5
device(config-if)# port-channel min-links 3
device# show etherchannel 25 summary
device# end

```

When the minimum links requirement is not met in standalone switches, the port-channel is flagged and assigned SM/SN or RM/RN state.

```

device# show etherchannel 5 summary

Flags: D - down P - bundled in port-channel
I - stand-alone s - suspended
H - Hot-standby (LACP only)
R - Layer3 S - Layer2
U - in use N- not in use, no aggregation
f - failed to allocate aggregator
M - not in use, no aggregation due to minimum links not met
m- not in use, port not aggregated due to minimum links not met
u - unsuitable for bundling
w - waiting to be aggregated
d - default port
Number of channel-groups in use: 6
Number of aggregators: 6

Group  Port-channel  Protocol  Ports
-----+-----+-----+-----
6      Po25(RM)      LACP      Gi1/3/1(D) Gi1/3/2(D) Gi2/2/25(D) Gi2/2/26(W)

```

Example: Configuring LACP Fast Rate Timer

This example shows you how to configure the LACP rate:

```

device> enable
device# configure terminal
device(config)# interface gigabitEthernet 2/1
device(config-if)# lacp rate fast
device(config-if)# exit
device(config)# end
device# show lacp internal
device# show lacp counters

```

The following is sample output from the **show lacp internal** command:

```

device# show lacp internal
Flags: S - Device is requesting Slow LACPDUs
F - Device is requesting Fast LACPDUs
A - Device is in Active mode P - Device is in Passive mode
Channel group 25
LACP port Admin Oper Port Port
Port Flags State Priority Key Key Number State

```

```

Tel/49 FA bndl 32768 0x19 0x19 0x32 0x3F
Tel/50 FA bndl 32768 0x19 0x19 0x33 0x3F
Tel/51 FA bndl 32768 0x19 0x19 0x34 0x3F
Tel/52 FA bndl 32768 0x19 0x19 0x35 0x3F

```

The following is sample output from the **show lacp counters** command:

```

device# show lacp counters

LACPDUs Marker Marker Response LACPDUs
Port Sent Recv Sent Recv Sent Recv Pkts Err
-----
Channel group: 24
Tel/1/27 2 2 0 0 0 0 0
Te2/1/25 2 2 0 0 0 0 0

```

Additional References for EtherChannels

Related Documents

Related Topic	Document Title
Layer 2 command reference	<i>Catalyst 2960-XR Switch Layer 2 Command Reference</i>

Standards and RFCs

Standard/RFC	Title
None	—

MIBs

MIB	MIBs Link
All the supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
<p>The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies.</p> <p>To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds.</p> <p>Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.</p>	http://www.cisco.com/support

Feature Information for EtherChannels

Release	Modification
Cisco IOS Release 15.0(2)EX1	This feature was introduced.
Cisco IOS 15.2(3)E2, Cisco IOS XE 3.7.2E	Auto-LAG feature was introduced.



CHAPTER 29

Configuring Link-State Tracking

- [Restrictions for Configuring Link-State Tracking, on page 515](#)
- [Understanding Link-State Tracking, on page 515](#)
- [How to Configure Link-State Tracking, on page 517](#)
- [Monitoring Link-State Tracking, on page 518](#)
- [Configuring Link-State Tracking: Example, on page 519](#)
- [Additional References for Link-State Tracking, on page 519](#)
- [Feature Information for Link-State Tracking, on page 520](#)

Restrictions for Configuring Link-State Tracking

- You can configure only two link-state groups per switch.
- An interface cannot be a member of more than one link-state group.
- An interface that is defined as an upstream interface in a link-state group cannot also be defined as a downstream interface in the link-state group.
- Do not enable link-state tracking on individual interfaces that will part of a downstream EtherChannel interface.

Understanding Link-State Tracking

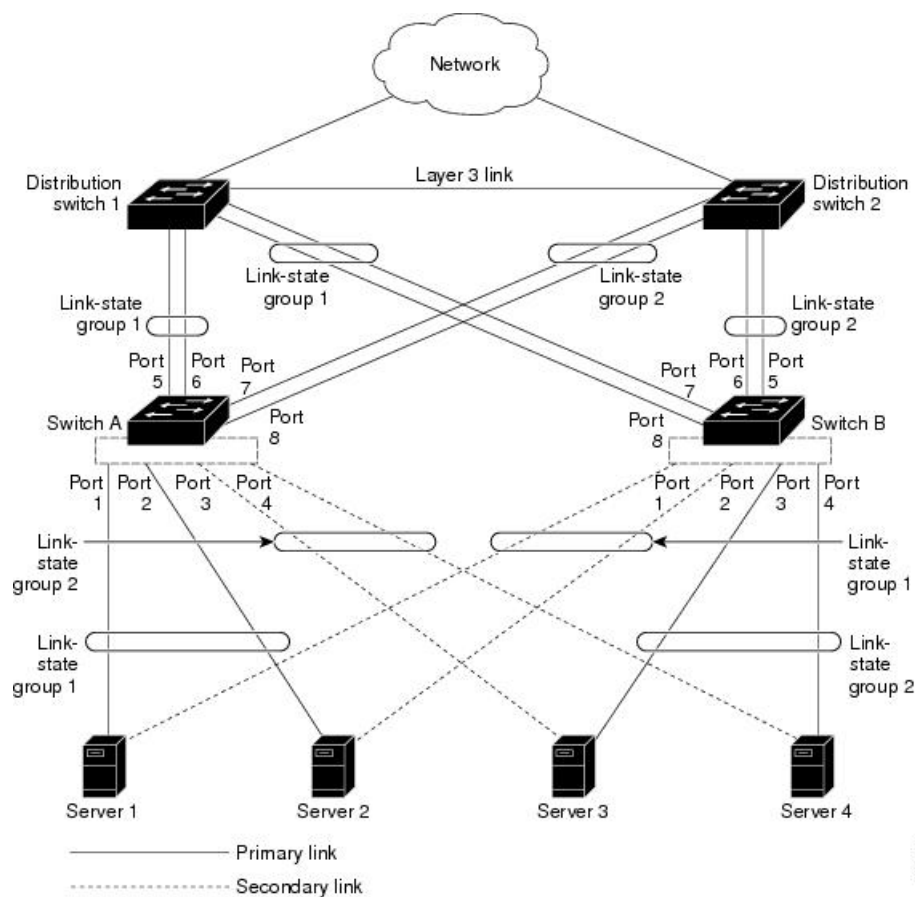
Link-state tracking, also known as trunk failover, binds the link state of multiple interfaces. Link-state tracking can be with server NIC adapter teaming to provide redundancy in the network. When the server NIC adapters are configured in a primary or secondary relationship, and the link is lost on the primary interface, network connectivity is transparently changed to the secondary interface.



Note An interface can be an aggregation of ports (an EtherChannel) or a single physical port in either access or trunk mode .

The configuration in this figure ensures that the network traffic flow is balanced.

Figure 51: Typical Link-State Tracking Configuration



- For links to switches and other network devices
 - Server 1 and server 2 use switch A for primary links and switch B for secondary links.
 - Server 3 and server 4 use switch B for primary links and switch A for secondary links.
- Link-state group 1 on switch A
 - Switch A provides primary links to server 1 and server 2 through link-state group 1. Port 1 is connected to server 1, and port 2 is connected to server 2. Port 1 and port 2 are the downstream interfaces in link-state group 1.
 - Port 5 and port 6 are connected to distribution switch 1 through link-state group 1. Port 5 and port 6 are the upstream interfaces in link-state group 1.
- Link-state group 2 on switch A
 - Switch A provides secondary links to server 3 and server 4 through link-state group 2. Port 3 is connected to server 3, and port 4 is connected to server 4. Port 3 and port 4 are the downstream interfaces in link-state group 2.
 - Port 7 and port 8 are connected to distribution switch 2 through link-state group 2. Port 7 and port 8 are the upstream interfaces in link-state group 2.

- Link-state group 2 on switch B
 - Switch B provides primary links to server 3 and server 4 through link-state group 2. Port 3 is connected to server 3, and port 4 is connected to server 4. Port 3 and port 4 are the downstream interfaces in link-state group 2.
 - Port 5 and port 6 are connected to distribution switch 2 through link-state group 2. Port 5 and port 6 are the upstream interfaces in link-state group 2.
- Link-state group 1 on switch B
 - Switch B provides secondary links to server 1 and server 2 through link-state group 1. Port 1 is connected to server 1, and port 2 is connected to server 2. Port 1 and port 2 are the downstream interfaces in link-state group 1.
 - Port 7 and port 8 are connected to distribution switch 1 through link-state group 1. Port 7 and port 8 are the upstream interfaces in link-state group 1.

In a link-state group, the upstream ports can become unavailable or lose connectivity because the distribution switch or router fails, the cables are disconnected, or the link is lost. These are the interactions between the downstream and upstream interfaces when link-state tracking is enabled:

- If any of the upstream interfaces are in the link-up state, the downstream interfaces can change to or remain in the link-up state.
- If all of the upstream interfaces become unavailable, link-state tracking automatically puts the downstream interfaces in the error-disabled state. Connectivity to and from the servers is automatically changed from the primary server interface to the secondary server interface. For example, in the previous figure, if the upstream link for port 6 is lost, the link states of downstream ports 1 and 2 do not change. However, if the link for upstream port 5 is also lost, the link state of the downstream ports changes to the link-down state. Connectivity to server 1 and server 2 is then changed from link-state group 1 to link-state group 2. The downstream ports 3 and 4 do not change state because they are in link-group 2.
- If the link-state group is configured, link-state tracking is disabled, and the upstream interfaces lose connectivity, the link states of the downstream interfaces remain unchanged. The server does not recognize that upstream connectivity has been lost and does not failover to the secondary interface.

You can recover a downstream interface link-down condition by removing the failed downstream port from the link-state group. To recover multiple downstream interfaces, disable the link-state group.

How to Configure Link-State Tracking

To enable link-state tracking, create a link-state group and specify the interfaces that are assigned to the group. This task is optional.

SUMMARY STEPS

1. **configure terminal**
2. **link state track** *number*
3. **interface** *interface-id*
4. **link state group** [*number*] {upstream | downstream}
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode.
Step 2	link state track number Example: Device(config)# <code>link state track 2</code>	Creates a link-state group and enables link-state tracking. The group number can be 1 or 2; the default is 1.
Step 3	interface interface-id Example: Device(config)# <code>interface gigabitethernet2/0/1</code>	Specifies a physical interface or range of interfaces to configure, and enters interface configuration mode. Valid interfaces include switch ports in access or trunk mode (IEEE 802.1q) or routed ports. Note Do not enable link-state tracking on individual interfaces that will be part of an Etherchannel interface.
Step 4	link state group [number]{<u>upstream</u> downstream} Example: Device(config-if)# <code>link state group 2 upstream</code>	Specifies a link-state group and configures the interface as either an upstream or downstream interface in the group.
Step 5	end Example: Device(config-if)# <code>end</code>	Returns to privileged EXEC mode.

Monitoring Link-State Tracking

You can display link-state tracking status using the command in this table.

Table 58: Commands for Monitoring Link-State Tracking Status

Command	Description
<code>show link state group [number] [detail]</code>	Displays the link-state group information.

Configuring Link-State Tracking: Example

This example shows how to create the link-state group 1 and configure the interfaces in the link-state group.

```
Device# configure terminal
Device(config)# link state track 1
Device(config-if)# interface range gigabitethernet1/0/21-22
Device(config-if)# link state group 1 upstream
Device(config-if)# interface gigabitethernet1/0/1
Device(config-if)# link state group 1 downstream
Device(config-if)# interface gigabitethernet1/0/3
Device(config-if)# link state group 1 downstream
Device(config-if)# interface gigabitethernet1/0/5
Device(config-if)# link state group 1 downstream
Device(config-if)# end
```

Additional References for Link-State Tracking

Related Documents

Related Topic	Document Title
Layer 2 command reference	<i>Catalyst 2960-XR Switch Layer 2 Command Reference</i>

Standards and RFCs

Standard/RFC	Title
None	—

MIBs

MIB	MIBs Link
All the supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
<p>The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies.</p> <p>To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds.</p> <p>Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.</p>	http://www.cisco.com/support

Feature Information for Link-State Tracking

Releases	Feature Information
Cisco IOS Release 15.0(2)EX1	This feature was introduced.



CHAPTER 30

Configuring Flex Links and the MAC Address-Table Move Update Feature

- [Restrictions for Configuring Flex Links and MAC Address-Table Move Update, on page 521](#)
- [Information About Flex Links and MAC Address-Table Move Update, on page 522](#)
- [How to Configure Flex Links and the MAC Address-Table Move Update Feature, on page 526](#)
- [Monitoring Flex Links, Multicast Fast Convergence, and MAC Address-Table Move Update, on page 531](#)
- [Configuration Examples for Flex Links, on page 532](#)
- [Additional References for Flex Links and MAC Address-Table Move Update, on page 536](#)
- [Feature Information for Flex Links and MAC Address-Table Move Update, on page 537](#)

Restrictions for Configuring Flex Links and MAC Address-Table Move Update

- Flex Links are supported only on Layer 2 ports and port channels.
- You can configure up to 16 backup links.
- You can configure only one Flex Links backup link for any active link, and it must be a different interface from the active interface.
- An interface can belong to only one Flex Links pair. An interface can be a backup link for only one active link. An active link cannot belong to another Flex Links pair.
- Neither of the links can be a port that belongs to an EtherChannel. However, you can configure two port channels (EtherChannel logical interfaces) as Flex Links, and you can configure a port channel and a physical interface as Flex Links, with either the port channel or the physical interface as the active link.
- A backup link does not have to be the same type (Gigabit Ethernet or port channel) as the active link. However, you should configure both Flex Links with similar characteristics so that there are no loops or changes in behavior if the standby link begins to forward traffic.
- STP is disabled on Flex Links ports. A Flex Links port does not participate in STP, even if the VLANs present on the port are configured for STP. When STP is not enabled, be sure that there are no loops in the configured topology.

Information About Flex Links and MAC Address-Table Move Update

Flex Links

Flex Links are a pair of a Layer 2 interfaces (device ports or port channels) where one interface is configured to act as a backup to the other. The feature provides an alternative solution to the Spanning Tree Protocol (STP). Users can disable STP and still retain basic link redundancy. Flex Links are typically configured in service provider or enterprise networks where customers do not want to run STP on the device. If the device is running STP, Flex Links are not necessary because STP already provides link-level redundancy or backup.

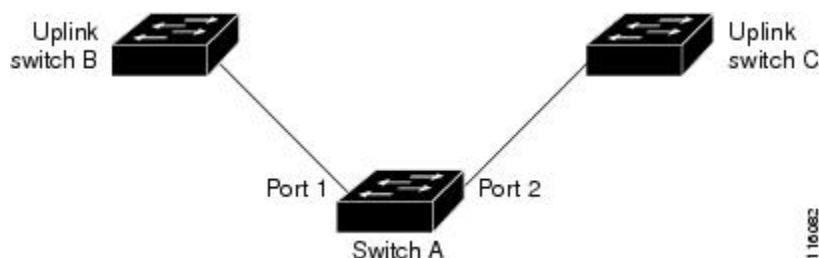
You configure Flex Links on one Layer 2 interface (the active link) by assigning another Layer 2 interface as the Flex Links or backup link. On devices, the Flex Links can be on the same device or on another device in the stack. When one of the links is up and forwarding traffic, the other link is in standby mode, ready to begin forwarding traffic if the other link shuts down. At any given time, only one of the interfaces is in the linkup state and forwarding traffic. If the primary link shuts down, the standby link starts forwarding traffic. When the active link comes back up, it goes into standby mode and does not forward traffic. STP is disabled on Flex Links interfaces.

Flex Links Configuration

In the following figure, ports 1 and 2 on device A are connected to uplink switches B and C. Because they are configured as Flex Links, only one of the interfaces is forwarding traffic; the other is in standby mode. If port 1 is the active link, it begins forwarding traffic between port 1 and switch B; the link between port 2 (the backup link) and switch C is not forwarding traffic. If port 1 goes down, port 2 comes up and starts forwarding traffic to switch C. When port 1 comes back up, it goes into standby mode and does not forward traffic; port 2 continues forwarding traffic.

You can also configure a preemption function, specifying the preferred port for forwarding traffic. For example, you can configure the Flex Links pair with preemption mode. In the scenario shown, when port 1 comes back up and has more bandwidth than port 2, port 1 begins forwarding traffic after 60 seconds. Port 2 becomes the standby port. You do this by entering the **switchport backup interface preemption mode bandwidth** and **switchport backup interface preemption delay** interface configuration commands.

Figure 52: Flex Links Configuration Example



If a primary (forwarding) link goes down, a trap notifies the network management stations. If the standby link goes down, a trap notifies the users.

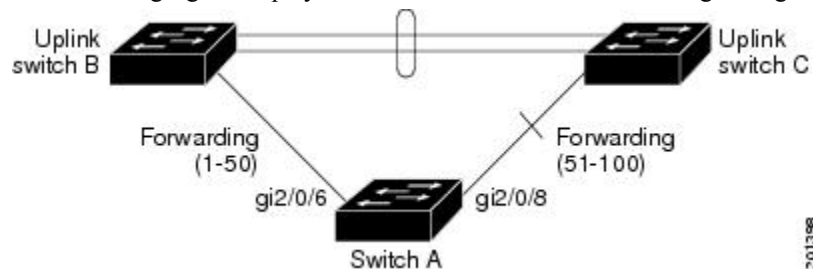
Flex Links are supported only on Layer 2 ports and port channels, not on VLANs or on Layer 3 ports.

VLAN Flex Links Load Balancing and Support

VLAN Flex Links load balancing allows users to configure a Flex Links pair so that both ports simultaneously forward the traffic for some mutually exclusive VLANs. For example, if Flex Links ports are configured for 1 to 100 VLANs, the traffic of the first 50 VLANs can be forwarded on one port and the rest on the other port. If one of the ports fail, the other active port forwards all the traffic. When the failed port comes back up, it resumes forwarding traffic in the preferred VLANs. In addition to providing the redundancy, this Flex Links pair can be used for load balancing. Flex Links VLAN load balancing does not impose any restrictions on uplink devices.

Figure 53: VLAN Flex Links Load-Balancing Configuration Example

The following figure displays a VLAN Flex Links load-balancing configuration.



Multicast Fast Convergence with Flex Links Failover

Multicast fast convergence reduces the multicast traffic convergence time after a Flex Links failure. Multicast fast convergence is implemented by a combination of learning the backup link as an mrouter port, generating IGMP reports, and leaking IGMP reports.

Learning the Other Flex Links Port as the mrouter Port

In a typical multicast network, there is a querier for each VLAN. A device deployed at the edge of a network has one of its Flex Links ports receiving queries. Flex Links ports are also always forwarding at any given time.

A port that receives queries is added as an mrouter port on the device. An mrouter port is part of all the multicast groups learned by the device. After a changeover, queries are received by the other Flex Links port. The other Flex Links port is then learned as the mrouter port. After changeover, multicast traffic then flows through the other Flex Links port. To achieve faster convergence of traffic, both Flex Links ports are learned as mrouter ports whenever either Flex Links port is learned as the mrouter port. Both Flex Links ports are always part of multicast groups.

Although both Flex Links ports are part of the groups in normal operation mode, all traffic on the backup port is blocked. The normal multicast data flow is not affected by the addition of the backup port as an mrouter port. When the changeover happens, the backup port is unblocked, allowing the traffic to flow. In this case, the upstream multicast data flows as soon as the backup port is unblocked.

Generating IGMP Reports

When the backup link comes up after the changeover, the upstream new distribution device does not start forwarding multicast data, because the port on the upstream router, which is connected to the blocked Flex Links port, is not part of any multicast group. The reports for the multicast groups were not forwarded by the

downstream device because the backup link is blocked. The data does not flow on this port, until it learns the multicast groups, which occurs only after it receives reports.

The reports are sent by hosts when a general query is received, and a general query is sent within 60 seconds in normal scenarios. When the backup link starts forwarding, to achieve faster convergence of multicast data, the downstream device immediately sends proxy reports for all the learned groups on this port without waiting for a general query.

Leaking IGMP Reports

To achieve multicast traffic convergence with minimal loss, a redundant data path must be set up before the Flex Links active link goes down. This can be achieved by leaking only IGMP report packets on the Flex Links backup link. These leaked IGMP report messages are processed by upstream distribution routers, so multicast data traffic gets forwarded to the backup interface. Because all incoming traffic on the backup interface is dropped at the ingress of the access device, no duplicate multicast traffic is received by the host. When the Flex Links active link fails, the access device starts accepting traffic from the backup link immediately. The only disadvantage of this scheme is that it consumes bandwidth on the link between the distribution devices and on the backup link between the distribution and access devices. This feature is disabled by default and can be configured by using the **switchport backup interface *interface-id* multicast fast-convergence** command.

When this feature has been enabled at changeover, the device does not generate the proxy reports on the backup port, which became the forwarding port.

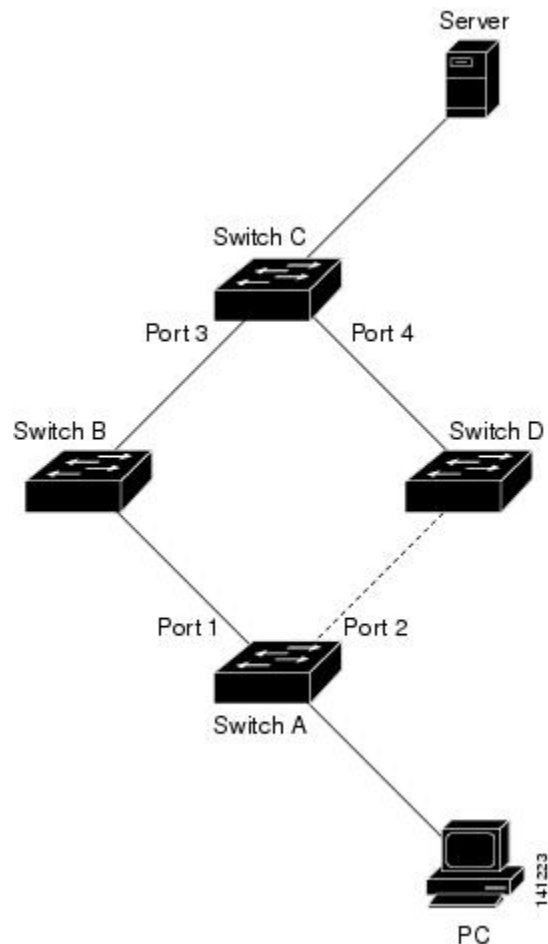
MAC Address-Table Move Update

The MAC address-table move update feature allows the device to provide rapid bidirectional convergence when a primary (forwarding) link goes down and the standby link begins forwarding traffic.

Figure 54: MAC Address-Table Move Update Example

In the following figure, switch A is an access switch, and ports 1 and 2 on switch A are connected to uplink devices B and D through a Flex Links pair. Port 1 is forwarding traffic, and port 2 is in the backup state. Traffic from the PC to the server is forwarded from port 1 to port 3. The MAC address of the PC has been

learned on port 3 of device C. Traffic from the server to the PC is forwarded from port 3 to port 1.



If the MAC address-table move update feature is not configured and port 1 goes down, port 2 starts forwarding traffic. However, for a short time, device C keeps forwarding traffic from the server to the PC through port 3, and the PC does not get the traffic because port 1 is down. If device C removes the MAC address of the PC on port 3 and relearns it on port 4, traffic can then be forwarded from the server to the PC through port 2.

If the MAC address-table move update feature is configured and enabled on the devices, and port 1 goes down, port 2 starts forwarding traffic from the PC to the server. The device sends a MAC address-table move update packet from port 2. Device C gets this packet on port 4 and immediately learns the MAC address of the PC on port 4, which reduces the reconvergence time.

You can configure the access device, device A, to *send* MAC address-table move update messages. You can also configure the uplink devices B, C, and D to *get* and process the MAC address-table move update messages. When device C gets a MAC address-table move update message from device A, device C learns the MAC address of the PC on port 4. Device C updates the MAC address table, including the forwarding table entry for the PC.

Device A does not need to wait for the MAC address-table update. The device detects a failure on port 1 and immediately starts forwarding server traffic from port 2, the new forwarding port. This change occurs in less than 100 milliseconds (ms). The PC is directly connected to device A, and the connection status does not change. Device A does not need to update the PC entry in the MAC address table.

Flex Links VLAN Load Balancing Configuration Guidelines

- For Flex Links VLAN load balancing, you must choose the preferred VLANs on the backup interface.
- You cannot configure a preemption mechanism and VLAN load balancing for the same Flex Links pair.

MAC Address-Table Move Update Configuration Guidelines

- You can enable and configure this feature on the access device to *send* the MAC address-table move updates.
- You can enable and configure this feature on the uplink devices to *get* the MAC address-table move updates.

Default Flex Links and MAC Address-Table Move Update Configuration

- Flex Links is not configured, and there are no backup interfaces defined.
- The preemption mode is off.
- The preemption delay is 35 seconds.
- The MAC address-table move update feature is not configured on the device.

How to Configure Flex Links and the MAC Address-Table Move Update Feature

Configuring Flex Links

SUMMARY STEPS

1. **configure terminal**
2. **interface *interface-id***
3. **switchport backup interface *interface-id***
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters global configuration mode.

	Command or Action	Purpose
Step 2	interface <i>interface-id</i> Example: Device(conf)# interface gigabitethernet1/0/1	Specifies the interface, and enters interface configuration mode. The interface can be a physical Layer 2 interface or a port channel (logical interface). The port-channel range is 1 to 48.
Step 3	switchport backup interface <i>interface-id</i> Example: Device(conf-if)# switchport backup interface gigabitethernet1/0/2	Configures a physical Layer 2 interface (or port channel) as part of a Flex Links pair with the interface. When one link is forwarding traffic, the other interface is in standby mode.
Step 4	end Example: Device(conf-if)# end	Returns to privileged EXEC mode.

Configuring a Preemption Scheme for a Pair of Flex Links

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **switchport backup interface** *interface-id*
4. **switchport backup interface** *interface-id* **preemption mode** [forced | bandwidth | off]
5. **switchport backup interface** *interface-id* **preemption delay** *delay-time*
6. **end**
7. **show interface** [*interface-id*] **switchport backup**
8. **copy running-config startup config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters global configuration mode
Step 2	interface <i>interface-id</i> Example: Device(conf)# interface gigabitethernet1/0/1	Specifies the interface, and enters interface configuration mode. The interface can be a physical Layer 2 interface or a port channel (logical interface). The port-channel range is 1 to 48.

	Command or Action	Purpose
Step 3	switchport backup interface <i>interface-id</i> Example: Device(conf-if)# switchport backup interface gigabitethernet1/0/2	Configures a physical Layer 2 interface (or port channel) as part of a Flex Links pair with the interface. When one link is forwarding traffic, the other interface is in standby mode.
Step 4	switchport backup interface <i>interface-id</i> preemption mode [forced bandwidth off] Example: Device(conf-if)# switchport backup interface gigabitethernet1/0/2 preemption mode forced	Configures a preemption mechanism and delay for a Flex Links interface pair. You can configure the preemption as: <ul style="list-style-type: none"> • forced—(Optional) The active interface always preempts the backup. • bandwidth—(Optional) The interface with the higher bandwidth always acts as the active interface. • off—(Optional) No preemption occurs from active to backup.
Step 5	switchport backup interface <i>interface-id</i> preemption delay <i>delay-time</i> Example: Device(conf-if)# switchport backup interface gigabitethernet1/0/2 preemption delay 50	Configures the time delay until a port preempts another port. Note Setting a delay time only works with forced and bandwidth modes.
Step 6	end Example: Device(conf-if)# end	Returns to privileged EXEC mode.
Step 7	show interface [<i>interface-id</i>] switchport backup Example: Device# show interface gigabitethernet1/0/2 switchport backup	Verifies the configuration.
Step 8	copy running-config startup config Example: Device# copy running-config startup config	(Optional) Saves your entries in the device startup configuration file.

Configuring VLAN Load Balancing on Flex Links

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **switchport backup interface** *interface-id* **prefer vlan** *vlan-range*
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 2	interface <i>interface-id</i> Example: Device (config)# interface gigabitethernet2/0/6	Specifies the interface, and enters interface configuration mode. The interface can be a physical Layer 2 interface or a port channel (logical interface). The port-channel range is 1 to 48.
Step 3	switchport backup interface <i>interface-id</i> prefer vlan <i>vlan-range</i> Example: Device (config-if)# switchport backup interface gigabitethernet2/0/8 prefer vlan 2	Configures a physical Layer 2 interface (or port channel) as part of a Flex Links pair with the interface and specifies the VLANs carried on the interface. The VLAN ID range is 1 to 4094.
Step 4	end Example: Device (config-if)# end	Returns to privileged EXEC mode.

Configuring MAC Address-Table Move Update

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. Use one of the following:
 - **switchport backup interface** *interface-id*
 - **switchport backup interface** *interface-id* **mmu primary vlan** *vlan-id*

4. `end`
5. `mac address-table move update transmit`
6. `end`

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p>configure terminal</p> <p>Example:</p> <pre>Device# configure terminal</pre>	Enters global configuration mode.
Step 2	<p>interface <i>interface-id</i></p> <p>Example:</p> <pre>Device#interface gigabitethernet1/0/1</pre>	Specifies the interface, and enters interface configuration mode. The interface can be a physical Layer 2 interface or a port channel (logical interface). The port-channel range is 1 to 48.
Step 3	<p>Use one of the following:</p> <ul style="list-style-type: none"> • switchport backup interface <i>interface-id</i> • switchport backup interface <i>interface-id</i> mmu primary vlan <i>vlan-id</i> <p>Example:</p> <pre>Device(config-if)# switchport backup interface gigabitethernet0/2 mmu primary vlan 2</pre>	<p>Configures a physical Layer 2 interface (or port channel), as part of a Flex Links pair with the interface. The MAC address-table move update VLAN is the lowest VLAN ID on the interface.</p> <p>Configure a physical Layer 2 interface (or port channel) and specifies the VLAN ID on the interface, which is used for sending the MAC address-table move update.</p> <p>When one link is forwarding traffic, the other interface is in standby mode.</p>
Step 4	<p>end</p> <p>Example:</p> <pre>Device(config-if)# end</pre>	Returns to global configuration mode.
Step 5	<p>mac address-table move update transmit</p> <p>Example:</p> <pre>Device(config)# mac address-table move update transmit</pre>	<p>Enables the access device to send MAC address-table move updates to other devices in the network if the primary link goes down and the device starts forwarding traffic through the standby link.</p> <p>Enter command mac address-table move update on the device, for MMU packets to update MAC tables. When the primary link comes back up, the MAC tables need to reconverge and this command will transmit the MMU, that will establish the behavior.</p>
Step 6	<p>end</p> <p>Example:</p>	Returns to privileged EXEC mode.

	Command or Action	Purpose
	Device(config)# end	

Configuring a Device to Obtain and Process MAC Address-Table Move Update Messages

SUMMARY STEPS

1. **configure terminal**
2. **mac address-table move update receive**
3. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters global configuration mode
Step 2	mac address-table move update receive Example: Device (config)# mac address-table move update receive	Enables the device to obtain and processes the MAC address-table move updates.
Step 3	end Example: Device (config)# end	Returns to privileged EXEC mode.

Monitoring Flex Links, Multicast Fast Convergence, and MAC Address-Table Move Update

Command	Purpose
show interface [<i>interface-id</i>] switchport backup	Displays the Flex Links backup interface configured for an interface or all the configured Flex Links and the state of each active and backup interface (up or standby mode).
show ip igmp profile address-table move update <i>profile-id</i>	Displays the specified IGMP profile or all the IGMP profiles defined on the device.

Command	Purpose
<code>show mac address-table move update</code>	Displays the MAC address-table move update information on the device.

Configuration Examples for Flex Links

Configuring Flex Links: Examples

This example shows how to verify the configuration after you configure an interface with a backup interface:

```
Device# show interface switchport backup

Switch Backup Interface Pairs:
Active Interface Backup Interface State
-----
GigabitEthernet1/0/1 GigabitEthernet1/0/2 Active Up/Backup Standby
```

This example shows how to verify the configuration after you configure the preemption mode as forced for a backup interface pair:

```
Device# show interface switchport backup detail

Switch Backup Interface Pairs:

Active Interface Backup Interface State
-----
GigabitEthernet1/0/211 GigabitEthernet1/0/2 Active Up/Backup Standby
Interface Pair : Gi1/0/1, Gi1/0/2
Preemption Mode : forced
Preemption Delay : 50 seconds
Bandwidth : 100000 Kbit (Gi1/0/1), 100000 Kbit (Gi1/0/2)
Mac Address Move Update Vlan : auto
```

Configuring VLAN Load Balancing on Flex Links: Examples

In the following example, VLANs 1 to 50, 60, and 100 to 120 are configured on the device:

```
Device(config)# interface gigabitethernet 2/0/6
Device(config-if)# switchport backup interface gigabitethernet 2/0/8 prefer vlan 60,100-120
```

When both interfaces are up, Gi2/0/8 forwards traffic for VLANs 60 and 100 to 120 and Gi2/0/6 forwards traffic for VLANs 1 to 50.

```
Device# show interfaces switchport backup

Switch Backup Interface Pairs:

Active Interface          Backup Interface          State
-----
-----
```

```
GigabitEthernet2/0/6    GigabitEthernet2/0/8    Active Up/Backup Standby
Vlans Preferred on Active Interface: 1-50
Vlans Preferred on Backup Interface: 60, 100-120
```

When a Flex Links interface goes down (LINK_DOWN), VLANs preferred on this interface are moved to the peer interface of the Flex Links pair. In this example, if interface Gi2/0/6 goes down, Gi2/0/8 carries all VLANs of the Flex Links pair.

```
Device# show interfaces switchport backup

Switch Backup Interface Pairs:

Active Interface      Backup Interface      State
-----
GigabitEthernet2/0/6  GigabitEthernet2/0/8  Active Down/Backup Up

Vlans Preferred on Active Interface: 1-50
Vlans Preferred on Backup Interface: 60, 100-120
```

When a Flex Links interface comes up, VLANs preferred on this interface are blocked on the peer interface and moved to the forwarding state on the interface that has just come up. In this example, if interface Gi2/0/6 comes up, VLANs preferred on this interface are blocked on the peer interface Gi2/0/8 and forwarded on Gi2/0/6.

```
Device# show interfaces switchport backup

Switch Backup Interface Pairs:

Active Interface      Backup Interface      State
-----
GigabitEthernet2/0/6  GigabitEthernet2/0/8  Active Up/Backup Standby

Vlans Preferred on Active Interface: 1-50
Vlans Preferred on Backup Interface: 60, 100-120
```

```
Device# show interfaces switchport backup detail

Switch Backup Interface Pairs:

Active Interface      Backup Interface      State
-----
FastEthernet1/0/3     FastEthernet1/0/4    Active Down/Backup Up

Vlans Preferred on Active Interface: 1-2,5-4094
Vlans Preferred on Backup Interface: 3-4
Preemption Mode      : off
Bandwidth : 10000 Kbit (Fa1/0/3), 100000 Kbit (Fa1/0/4)
Mac Address Move Update Vlan : auto
```

Configuring the MAC Address-Table Move Update: Examples

This example shows how to verify the configuration after you configure an access device to send MAC address-table move updates:

```
Device# show mac address-table move update
```

```

Switch-ID : 010b.4630.1780
Dst mac-address : 0180.c200.0010
Vlans/Macs supported : 1023/8320
Default/Current settings: Rcv Off/On, Xmt Off/On
Max packets per min : Rcv 40, Xmt 60
Rcv packet count : 5
Rcv conforming packet count : 5
Rcv invalid packet count : 0
Rcv packet count this min : 0
Rcv threshold exceed count : 0
Rcv last sequence# this min : 0
Rcv last interface : Po2
Rcv last src-mac-address : 000b.462d.c502
Rcv last switch-ID : 0403.fd6a.8700
Xmt packet count : 0
Xmt packet count this min : 0
Xmt threshold exceed count : 0
Xmt pak buf unavail cnt : 0
Xmt last interface : None

```

Configuring Multicast Fast Convergence with Flex Links Failover: Examples

These are configuration examples for learning the other Flex Links port as the mrouter port when Flex Links is configured on GigabitEthernet1/0/11 and GigabitEthernet1/0/12, and output for the **show interfaces switchport backup** command:

```

Device# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Device(config)# interface GigabitEthernet1/0/11
Device(config-if)# switchport trunk encapsulation dot1q
Device(config-if)# switchport mode trunk
Device(config-if)# switchport backup interface GigabitEthernet1/0/12
Device(config-if)# exit
Device(config)# interface GigabitEthernet1/0/12
Device(config-if)# switchport trunk encapsulation dot1q
Device(config-if)# switchport mode trunk
Device(config-if)# end
Device# show interfaces switchport backup detail
Switch Backup Interface Pairs:
Active Interface Backup Interface State
GigabitEthernet1/0/11 GigabitEthernet1/0/12 Active Up/Backup Standby
Preemption Mode : off
Multicast Fast Convergence : Off
Bandwidth : 100000 Kbit (Gi1/0/11), 100000 Kbit (Gi1/0/12)
Mac Address Move Update Vlan : auto

```

This output shows a querier for VLANs 1 and 401, with their queries reaching the device through GigabitEthernet1/0/11:

```

Device# show ip igmp snooping querier

```

Vlan	IP Address	IGMP Version	Port
1	0.0.0.10	v2	Gi1/0/11
401	41.41.41.1	v2	Gi1/0/11

This example is output for the **show ip igmp snooping mrouter** command for VLANs 1 and 401:

```
Device# show ip igmp snooping mrouter

Vlan  ports
----  -
1     Gi1/0/11 (dynamic), Gi1/0/12 (dynamic)
401   Gi1/0/11 (dynamic), Gi1/0/12 (dynamic)
```

Similarly, both Flex Links ports are part of learned groups. In this example, GigabitEthernet2/0/11 is a receiver/host in VLAN 1, which is interested in two multicast groups:

```
Device# show ip igmp snooping groups

Vlan  Group    Type   Version  Port List
-----
1     228.1.5.1  igmp   v2       Gi1/0/11, Gi1/0/12, Gi2/0/11
1     228.1.5.2  igmp   v2       Gi1/0/11, Gi1/0/12, Gi2/0/11
```

When a host responds to the general query, the device forwards this report on all the mrouter ports. In this example, when a host sends a report for the group 228.1.5.1, it is forwarded only on GigabitEthernet1/0/11, because the backup port GigabitEthernet1/0/12 is blocked. When the active link, GigabitEthernet1/0/11, goes down, the backup port, GigabitEthernet1/0/12, begins forwarding.

As soon as this port starts forwarding, the device sends proxy reports for the groups 228.1.5.1 and 228.1.5.2 on behalf of the host. The upstream router learns the groups and starts forwarding multicast data. This is the default behavior of Flex Links. This behavior changes when the user configures fast convergence using the **switchport backup interface gigabitEthernet 1/0/12 multicast fast-convergence** command. This example shows turning on this feature:

```
Device# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Device(config)# interface gigabitEthernet 1/0/11
Device(config-if)# switchport backup interface gigabitEthernet 1/0/12 multicast
fast-convergence
Device(config-if)# exit
Device# show interfaces switchport backup detail

Switch Backup Interface Pairs:
Active      Interface      Backup Interface State
-----
GigabitEthernet1/0/11 GigabitEthernet1/0/12 Active Up/Backup Standby
Preemption Mode : off
Multicast Fast Convergence : On
Bandwidth : 100000 Kbit (Gi1/0/11), 100000 Kbit (Gi1/0/12)
Mac Address Move Update Vlan : auto
```

This output shows a querier for VLAN 1 and 401 with their queries reaching the device through GigabitEthernet1/0/11:

```
Device# show ip igmp snooping querier

Vlan  IP Address  IGMP Version  Port
-----
1     10.0.0.10   v2            Gi1/0/11
```

```
401 41.41.41.1 v2 Gi1/0/11
```

This is output for the **show ip igmp snooping mrouter** command for VLAN 1 and 401:

```
Device# show ip igmp snooping mrouter

Vlan    ports
----    -
1       Gi1/0/11(dynamic), Gi1/0/12(dynamic)
401     Gi1/0/11(dynamic), Gi1/0/12(dynamic)
```

Similarly, both the Flex Links ports are a part of the learned groups. In this example, GigabitEthernet2/0/11 is a receiver/host in VLAN 1, which is interested in two multicast groups:

```
Device# show ip igmp snooping groups

Vlan  Group    Type    Version    Port List
-----
1     228.1.5.1  igmp    v2         Gi1/0/11, Gi1/0/12, Gi2/0/11
1     228.1.5.2  igmp    v2         Gi1/0/11, Gi1/0/12, Gi2/0/11
```

Whenever a host responds to the general query, the device forwards this report on all the mrouter ports. When you turn on this feature through the command-line port, and when a report is forwarded by the device on GigabitEthernet1/0/11, it is also leaked to the backup port GigabitEthernet1/0/12. The upstream router learns the groups and starts forwarding multicast data, which is dropped at the ingress because GigabitEthernet1/0/12 is blocked. When the active link, GigabitEthernet1/0/11, goes down, the backup port, GigabitEthernet1/0/12, begins forwarding. You do not need to send any proxy reports as the multicast data is already being forwarded by the upstream router. By leaking reports to the backup port, a redundant multicast path has been set up, and the time taken for the multicast traffic convergence is very minimal.

Additional References for Flex Links and MAC Address-Table Move Update

Related Documents

Related Topic	Document Title
Layer 2 command reference	<i>Catalyst 2960-XR Switch Layer 2 Command Reference</i>
switchport backup interface command	<i>Catalyst 2960-XR Switch Interface and Hardware Component Command Reference</i>

Error Message Decoder

Description	Link
To help you research and resolve system error messages in this release, use the Error Message Decoder tool.	https://www.cisco.com/cgi-bin/Support/Errordecoder/index.cgi

Standards and RFCs

Standard/RFC	Title
None	—

MIBs

MIB	MIBs Link
All the supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
<p>The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies.</p> <p>To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds.</p> <p>Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.</p>	http://www.cisco.com/support

Feature Information for Flex Links and MAC Address-Table Move Update

Release	Modification
Cisco IOS Release 15.0(2)EX1	This feature was introduced.



CHAPTER 31

Configuring UniDirectional Link Detection

- [Restrictions for Configuring UDLD, on page 539](#)
- [Information About UDLD, on page 539](#)
- [How to Configure UDLD, on page 542](#)
- [Monitoring and Maintaining UDLD, on page 544](#)
- [Additional References for UDLD, on page 544](#)
- [Feature Information for UDLD, on page 545](#)

Restrictions for Configuring UDLD

The following are restrictions for configuring UniDirectional Link Detection (UDLD):

- A UDLD-capable port cannot detect a unidirectional link if it is connected to a UDLD-incapable port of another device.
- When configuring the mode (normal or aggressive), make sure that the same mode is configured on both sides of the link.



Caution

Loop guard works only on point-to-point links. We recommend that each end of the link has a directly connected device that is running STP.

Information About UDLD

UniDirectional Link Detection (UDLD) is a Layer 2 protocol that enables devices connected through fiber-optic or twisted-pair Ethernet cables to monitor the physical configuration of the cables and detect when a unidirectional link exists. All connected devices must support UDLD for the protocol to successfully identify and disable unidirectional links. When UDLD detects a unidirectional link, it disables the affected port and alerts you. Unidirectional links can cause a variety of problems, including spanning-tree topology loops.

Modes of Operation

UDLD two modes of operation: normal (the default) and aggressive. In normal mode, UDLD can detect unidirectional links due to misconnected ports on fiber-optic connections. In aggressive mode, UDLD can

also detect unidirectional links due to one-way traffic on fiber-optic and twisted-pair links and to misconnected ports on fiber-optic links.

In normal and aggressive modes, UDLD works with the Layer 1 mechanisms to learn the physical status of a link. At Layer 1, autonegotiation takes care of physical signaling and fault detection. UDLD performs tasks that autonegotiation cannot perform, such as detecting the identities of neighbors and shutting down misconnected ports. When you enable both autonegotiation and UDLD, the Layer 1 and Layer 2 detections work together to prevent physical and logical unidirectional connections and the malfunctioning of other protocols.

A unidirectional link occurs whenever traffic sent by a local device is received by its neighbor but traffic from the neighbor is not received by the local device.

Normal Mode

In normal mode, UDLD detects a unidirectional link when fiber strands in a fiber-optic port are misconnected and the Layer 1 mechanisms do not detect this misconnection. If the ports are connected correctly but the traffic is one way, UDLD does not detect the unidirectional link because the Layer 1 mechanism, which is supposed to detect this condition, does not do so. In this case, the logical link is considered undetermined, and UDLD does not disable the port.

When UDLD is in normal mode, if one of the fiber strands in a pair is disconnected, as long as autonegotiation is active, the link does not stay up because the Layer 1 mechanisms detects a physical problem with the link. In this case, UDLD does not take any action and the logical link is considered undetermined.

Aggressive Mode

In aggressive mode, UDLD detects a unidirectional link by using the previous detection methods. UDLD in aggressive mode can also detect a unidirectional link on a point-to-point link on which no failure between the two devices is allowed. It can also detect a unidirectional link when one of these problems exists:

- On fiber-optic or twisted-pair links, one of the ports cannot send or receive traffic.
- On fiber-optic or twisted-pair links, one of the ports is down while the other is up.
- One of the fiber strands in the cable is disconnected.

In these cases, UDLD disables the affected port.

In a point-to-point link, UDLD hello packets can be considered as a heart beat whose presence guarantees the health of the link. Conversely, the loss of the heart beat means that the link must be shut down if it is not possible to reestablish a bidirectional link.

If both fiber strands in a cable are working normally from a Layer 1 perspective, UDLD in aggressive mode detects whether those fiber strands are connected correctly and whether traffic is flowing bidirectionally between the correct neighbors. This check cannot be performed by autonegotiation because autonegotiation operates at Layer 1.

Methods to Detect Unidirectional Links

UDLD operates by using two methods:

- Neighbor database maintenance
- Event-driven detection and echoing

Neighbor Database Maintenance

UDLD learns about other UDLD-capable neighbors by periodically sending a hello packet (also called an advertisement or probe) on every active port to keep each device informed about its neighbors.

When the device receives a hello message, it caches the information until the age time (hold time or time-to-live) expires. If the device receives a new hello message before an older cache entry ages, the device replaces the older entry with the new one.

Whenever a port is disabled and UDLD is running, whenever UDLD is disabled on a port, or whenever the device is reset, UDLD clears all existing cache entries for the ports affected by the configuration change. UDLD sends at least one message to inform the neighbors to flush the part of their caches affected by the status change. The message is intended to keep the caches synchronized.

Event-Driven Detection and Echoing

UDLD relies on echoing as its detection operation. Whenever a UDLD device learns about a new neighbor or receives a resynchronization request from an out-of-sync neighbor, it restarts the detection window on its side of the connection and sends echo messages in reply. Because this behavior is the same on all UDLD neighbors, the sender of the echoes expects to receive an echo in reply.

If the detection window ends and no valid reply message is received, the link might shut down, depending on the UDLD mode. When UDLD is in normal mode, the link might be considered undetermined and might not be shut down. When UDLD is in aggressive mode, the link is considered unidirectional, and the port is disabled.

UDLD Reset Options

If an interface becomes disabled by UDLD, you can use one of the following options to reset UDLD:

- The **udld reset** interface configuration command.
- The **shutdown** interface configuration command followed by the **no shutdown** interface configuration command restarts the disabled port.
- The **no udld {aggressive | enable}** global configuration command followed by the **udld {aggressive | enable}** global configuration command reenables the disabled ports.
- The **no udld port** interface configuration command followed by the **udld port [aggressive]** interface configuration command reenables the disabled fiber-optic port.
- The **errdisable recovery cause udld** global configuration command enables the timer to automatically recover from the UDLD error-disabled state, and the **errdisable recovery interval interval** global configuration command specifies the time to recover from the UDLD error-disabled state.

Default UDLD Configuration

Table 59: Default UDLD Configuration

Feature	Default Setting
UDLD global enable state	Globally disabled
UDLD per-port enable state for fiber-optic media	Disabled on all Ethernet fiber-optic ports

Feature	Default Setting
UDLD per-port enable state for twisted-pair (copper) media	Disabled on all Ethernet 10/100 and 1000BASE-TX ports
UDLD aggressive mode	Disabled

How to Configure UDLD

Enabling UDLD Globally

Follow these steps to enable UDLD in the aggressive or normal mode and to set the configurable message timer on all fiber-optic ports on the device.

SUMMARY STEPS

1. **configure terminal**
2. **udld {aggressive | enable | message time *message-timer-interval*}**
3. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 2	udld {aggressive enable message time <i>message-timer-interval</i>} Example: Device(config)# udld enable message time 10	Specifies the UDLD mode of operation: <ul style="list-style-type: none"> • aggressive—Enables UDLD in aggressive mode on all fiber-optic ports. • enable—Enables UDLD in normal mode on all fiber-optic ports on the device. UDLD is disabled by default. An individual interface configuration overrides the setting of the udld enable global configuration command. • message time <i>message-timer-interval</i>—Configures the period of time between UDLD probe messages on ports that are in the advertisement phase and are detected to be bidirectional. The range is from 1 to 90 seconds; the default value is 15.

	Command or Action	Purpose
		<p>Note This command affects fiber-optic ports only. Use the udld interface configuration command to enable UDLD on other port types.</p> <p>Use the no form of this command, to disable UDLD.</p>
Step 3	<p>end</p> <p>Example:</p> <pre>Device(config)# end</pre>	Returns to privileged EXEC mode.

Enabling UDLD on an Interface

Follow these steps either to enable UDLD in the aggressive or normal mode or to disable UDLD on a port.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **udld port** [aggressive]
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p>configure terminal</p> <p>Example:</p> <pre>Device# configure terminal</pre>	Enters global configuration mode.
Step 2	<p>interface <i>interface-id</i></p> <p>Example:</p> <pre>Device(config)# interface gigabitethernet 1/0/1</pre>	Specifies the port to be enabled for UDLD, and enters interface configuration mode.
Step 3	<p>udld port [aggressive]</p> <p>Example:</p> <pre>Device(config-if)# udld port aggressive</pre>	<p>UDLD is disabled by default.</p> <ul style="list-style-type: none"> • udld port—Enables UDLD in normal mode on the specified port. • udld port aggressive—(Optional) Enables UDLD in aggressive mode on the specified port.

	Command or Action	Purpose
		Note Use the no udld port interface configuration command to disable UDLD on a specified fiber-optic port.
Step 4	end Example: Device(config-if)# end	Returns to privileged EXEC mode.

Monitoring and Maintaining UDLD

Command	Purpose
show udld [<i>interface-id</i> neighbors]	Displays the UDLD status for the specified port or for all ports.

Additional References for UDLD

Related Documents

Related Topic	Document Title
For complete syntax and usage information for the commands used in this chapter.	<i>Catalyst 2960-XR Switch Layer 2 Command Reference</i>

Standards and RFCs

Standard/RFC	Title
None	—

MIBs

MIB	MIBs Link
All the supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
<p>The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies.</p> <p>To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds.</p> <p>Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.</p>	http://www.cisco.com/support

Feature Information for UDLD

Release	Modification
Cisco IOS Release 15.0(2)EX1	This feature was introduced.



CHAPTER 32

Configuring Resilient Ethernet Protocol

- [Overview of Resilient Ethernet Protocol, on page 547](#)
- [How to Configure Resilient Ethernet Protocol, on page 552](#)
- [Monitoring Resilient Ethernet Protocol Configurations, on page 560](#)
- [Configuration Examples for Resilient Ethernet Protocol, on page 562](#)

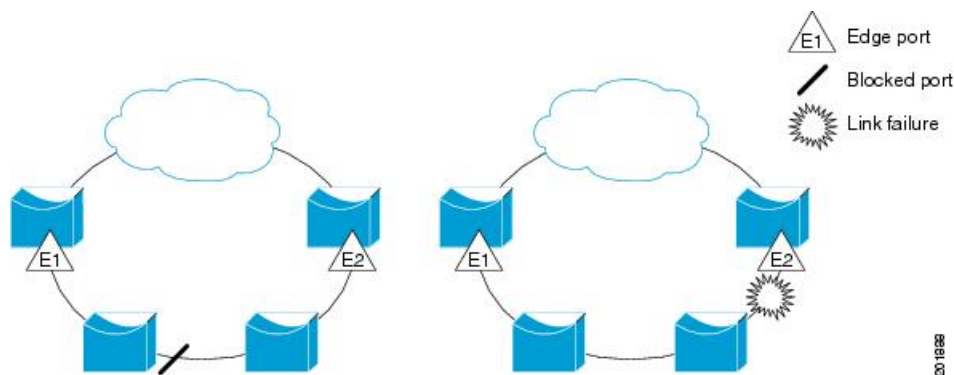
Overview of Resilient Ethernet Protocol

Resilient Ethernet Protocol (REP) is a Cisco-proprietary protocol that provides an alternative to Spanning Tree Protocol (STP) to control network loops, handle link failures, and improve convergence time. REP controls a group of ports connected in a segment, ensures that the segment does not create any bridging loops, and responds to link failures within the segment. REP provides a basis for constructing more complex networks and supports VLAN load balancing.

A REP segment is a chain of ports connected to each other and configured with a segment ID. Each segment consists of standard (nonedge) segment ports and two user-configured edge ports. A device can have no more than two ports that belong to the same segment, and each segment port can have only one external neighbor. A segment can go through a shared medium, but on any link, only two ports can belong to the same segment. REP is supported only on Trunk Ethernet Flow Point (EFP) interfaces.

The following figure shows an example of a segment consisting of six ports spread across four switches. Ports E1 and E2 are configured as edge ports. When all the ports are operational (as in the segment on the left), a single port is blocked, as shown by the diagonal line. When there is a failure in the network, the blocked port returns to the forwarding state to minimize network disruption.

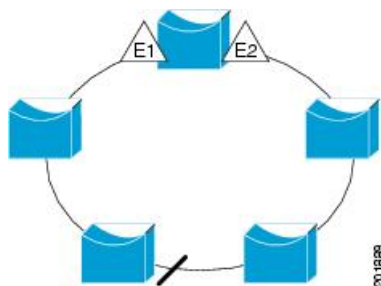
Figure 55: REP Open Segment



The segment shown in the figure above is an open segment; there is no connectivity between the two edge ports. The REP segment cannot cause a bridging loop, and you can safely connect the segment edges to any network. All the hosts connected to devices inside the segment have two possible connections to the rest of the network through the edge ports, but only one connection is accessible at any time. If a failure occurs on any segment or on any port on a REP segment, REP unblocks all the ports to ensure that connectivity is available through the other gateway.

The segment shown in the following figure is a ring segment, with both the edge ports located on the same device. With this configuration, you can create a redundant connection between any two devices in the segment.

Figure 56: REP Ring Segment



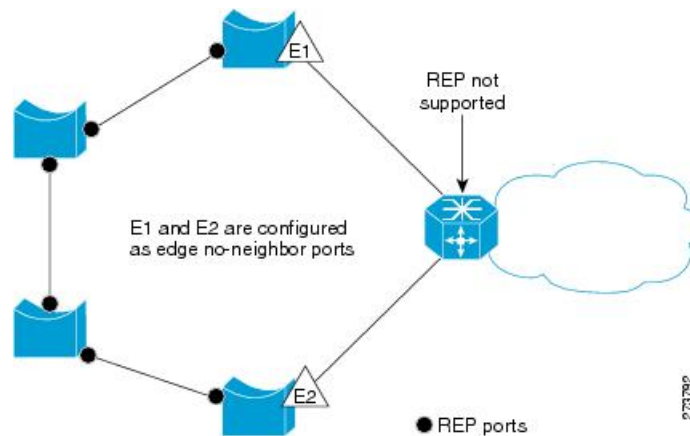
REP segments have the following characteristics:

- If all the ports in a segment are operational, one port (referred to as the *alternate* port) is in the blocked state for each VLAN. If VLAN load balancing is configured, two ports in the segment control the blocked state of VLANs.
- If one or more ports in a segment is not operational, and cause a link failure, all the ports forward traffic on all the VLANs to ensure connectivity.
- In case of a link failure, alternate ports are unblocked as quickly as possible. When the failed link is up, a logically blocked port per VLAN is selected with minimal disruption to the network.

You can construct almost any type of network based on REP segments. REP also supports VLAN load balancing, which is controlled by the primary edge port (any port in the segment).

In access ring-topologies, the neighboring switch might not support REP as shown in the following figure. In this scenario, you can configure the non-REP-facing ports (E1 and E2) as edge no-neighbor ports. These ports inherit all the properties of edge ports, and you can configure them the same as any edge port, including configuring them to send STP or REP topology change notices to the aggregation switch. In this scenario, the STP topology change notice (TCN) that is sent is a multiple spanning-tree (MST) STP message.

Figure 57: Edge No-Neighbor Ports



REP has these limitations:

- You must configure each segment port; an incorrect configuration might cause forwarding loops in the networks.
- REP can manage only a single failed port within the segment; multiple port failures within the REP segment cause loss of network connectivity.
- You should configure REP only in networks with redundancy. Configuring REP in a network without redundancy causes loss of connectivity.

Link Integrity

REP does not use an end-to-end polling function between edge ports to verify link integrity. It implements local link failure detection. The REP Link Status Layer (LSL) detects its REP-aware neighbor and establishes connectivity within the segment. All the VLANs are blocked on an interface until the neighbor is detected. After the neighbor is identified, REP determines which neighbor port should become the alternate port and which ports should forward traffic.

Each port in a segment has a unique port ID. The port ID format is similar to that used by the spanning tree algorithm: a port number (unique on the bridge) associated to a MAC address (unique in the network). When a segment port is coming up, its LSL starts sending packets that include the segment ID and the port ID. The port is declared as operational after it performs a three-way handshake with a neighbor in the same segment.

A segment port does not become operational if:

- No neighbor has the same segment ID.
- More than one neighbor has the same segment ID.
- A neighbor does not acknowledge a local port as a peer.

Each port creates an adjacency with its immediate neighbor. After the neighbor adjacencies are created, the ports negotiate with each other to determine the blocked port for the segment, which will function as the alternate port. All the other ports become unblocked. By default, REP packets are sent to a bridge protocol data unit-class MAC address. The packets can also be sent to a Cisco multicast address, which is used only to send blocked port advertisement (BPA) messages when there is a failure in the segment. The packets are dropped by the devices not running REP.

Fast Convergence

REP runs on a physical link basis and not on a per-VLAN basis. Only one hello message is required for all the VLANs, and this reduces the load on the protocol. We recommend that you create VLANs consistently on all the switches in a given segment and configure the same allowed VLANs on the REP trunk ports. To avoid the delay introduced by relaying messages in software, REP also allows some packets to be flooded to a regular multicast address. These messages operate at the hardware flood layer (HFL) and are flooded to the entire network, not just the REP segment. Switches that do not belong to the segment treat them as data traffic. You can control flooding of these messages by configuring an administrative VLAN for the entire domain or for a particular segment.

The estimated convergence recovery time on fiber interfaces is between 50 ms and 200 ms for the local segment with 200 VLANs configured. Convergence for VLAN load balancing is 300 ms or less.

VLAN Load Balancing

One edge port in the REP segment acts as the primary edge port; and another as the secondary edge port. It is the primary edge port that always participates in VLAN load balancing in the segment. REP VLAN balancing is achieved by blocking some VLANs at a configured alternate port and all the other VLANs at the primary edge port. When you configure VLAN load balancing, you can specify the alternate port in one of three ways:

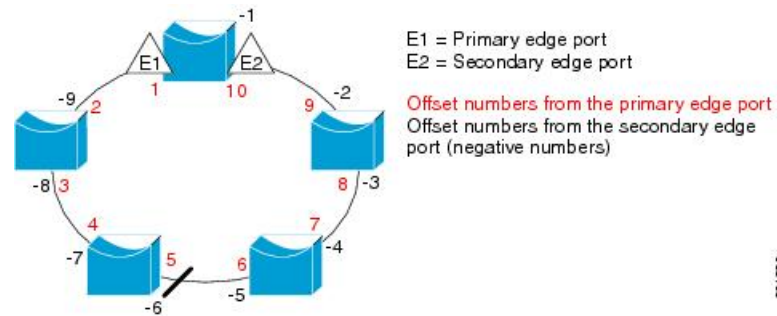
- By entering the port ID of the interface. To identify the port ID of a port in the segment, enter the **show interface rep detail** interface configuration command for the port.
- By entering the **preferred** keyword to select the port that you previously configured as the preferred alternate port with the **rep segment segment-id preferred** interface configuration command.
- By entering the neighbor offset number of a port in the segment, which identifies the downstream neighbor port of an edge port. The neighbor offset number range is -256 to +256; a value of 0 is invalid. The primary edge port has an offset number of 1; positive numbers above 1 identify downstream neighbors of the primary edge port. Negative numbers indicate the secondary edge port (offset number -1) and its downstream neighbors.



Note Configure offset numbers on the primary edge port by identifying a port's downstream position from the primary (or secondary) edge port. Never enter an offset value of 1 because that is the offset number of the primary edge port.

The following figure shows neighbor offset numbers for a segment, where E1 is the primary edge port and E2 is the secondary edge port. The numbers inside the ring are numbers offset from the primary edge port; the numbers outside of the ring show the offset numbers from the secondary edge port. Note that you can identify all the ports (except the primary edge port) by either a positive offset number (downstream position from the primary edge port) or a negative offset number (downstream position from the secondary edge port). If E2 became the primary edge port, its offset number would then be 1 and E1 would be -1.

Figure 58: Neighbor Offset Numbers in a Segment



When the REP segment is complete, all the VLANs are blocked. When you configure VLAN load balancing, you must also configure triggers in one of two ways:

- Manually trigger VLAN load balancing at any time by entering the **rep preempt segment** *segment-id* privileged EXEC command on the switch that has the primary edge port.
- Configure a preempt delay time by entering the **rep preempt delay** *seconds* interface configuration command. After a link failure and recovery, VLAN load balancing begins after the configured preemption time period elapses. Note that the delay timer restarts if another port fails before the time has elapsed.



Note When VLAN load balancing is configured, it does not start working until triggered by either manual intervention or a link failure and recovery.

When VLAN load balancing is triggered, the primary edge port sends out a message to alert all the interfaces in the segment about the preemption. When the secondary port receives the message, the message is sent to the network to notify the alternate port to block the set of VLANs specified in the message and to notify the primary edge port to block the remaining VLANs.

You can also configure a particular port in the segment to block all the VLANs. Only the primary edge port initiates VLAN load balancing, which is not possible if the segment is not terminated by an edge port on each end. The primary edge port determines the local VLAN load-balancing configuration.

Reconfigure the primary edge port to reconfigure load balancing. When you change the load-balancing configuration, the primary edge port waits for the **rep preempt segment** command or for the configured preempt delay period after a port failure and recovery, before executing the new configuration. If you change an edge port to a regular segment port, the existing VLAN load-balancing status does not change. Configuring a new edge port might cause a new topology configuration.

Spanning Tree Interaction

REP does not interact with the STP or the Flex Link feature, but can coexist with both. A port that belongs to a segment is removed from spanning tree control, and STP BPDUs are not accepted or sent from segment ports. Therefore, STP cannot run on a segment.

To migrate from an STP ring configuration to an REP segment configuration, begin by configuring a single port in the ring as part of the segment and continue by configuring contiguous ports to minimize the number of segments. Since each segment always contains a blocked port, multiple segments means multiple blocked

ports and a potential loss of connectivity. After the segment is configured in both directions up to the location of the edge ports, configure the edge ports.

REP Ports

REP segments consist of Failed, Open, or Alternate ports:

- A port configured as a regular segment port starts as a failed port.
- After the neighbor adjacencies are determined, the port transitions to alternate port state, blocking all the VLANs on the interface. Blocked-port negotiations occur, and when the segment settles, one blocked port remains in the alternate role and all the other ports become open ports.
- When a failure occurs in a link, all the ports move to the Failed state. When the Alternate port receives the failure notification, it changes to the Open state, forwarding all the VLANs.

A regular segment port converted to an edge port, or an edge port converted to a regular segment port, does not always result in a topology change. If you convert an edge port into a regular segment port, VLAN load balancing is not implemented unless it has been configured. For VLAN load balancing, you must configure two edge ports in the segment.

A segment port that is reconfigured as a spanning tree port restarts according to the spanning tree configuration. By default, this is a designated blocking port. If PortFast is configured or if STP is disabled, the port goes into the forwarding state.

How to Configure Resilient Ethernet Protocol

A segment is a collection of ports connected to one another in a chain and configured with a segment ID. To configure REP segments, configure the REP administrative VLAN (or use the default VLAN 1) and then add the ports to the segment, using interface configuration mode. You should configure two edge ports in a segment, with one of them being the primary edge port and the other the secondary edge port by default. A segment should have only one primary edge port. If you configure two ports in a segment as primary edge ports, for example, ports on different switches, the REP selects one of them to serve as the segment's primary edge port. If required, you can configure the location to which segment topology change notices (STCNs) and VLAN load balancing are to be sent.

Default REP Configuration

REP is disabled on all the interfaces. When enabled, the interface is a regular segment port unless it is configured as an edge port.

When REP is enabled, the task of sending segment topology change notices (STCNs) is disabled, all the VLANs are blocked, and the administrative VLAN is VLAN 1.

When VLAN load balancing is enabled, the default is manual preemption with the delay timer disabled. If VLAN load balancing is not configured, the default after manual preemption is to block all the VLANs in the primary edge port.

REP Configuration Guidelines

Follow these guidelines when configuring REP:

- We recommend that you begin by configuring one port and then configure contiguous ports to minimize the number of segments and the number of blocked ports.
- If more than two ports in a segment fail when no external neighbors are configured, one port goes into a forwarding state for the data path to help maintain connectivity during configuration. In the **show rep interface** command output, the Port Role for this port is displayed as **Fail Logical Open**; the Port Role for the other failed port is displayed as **Fail No Ext Neighbor**. When the external neighbors for the failed ports are configured, the ports go through the alternate port transitions and eventually go to an open state, or remain as the alternate port, based on the alternate port selection mechanism.
- REP ports must be Layer 2 IEEE 802.1Q or Trunk ports.
- We recommend that you configure all the trunk ports in a segment with the same set of allowed VLANs.
- Be careful when configuring REP through a Telnet connection because REP blocks all the VLANs until another REP interface sends a message to unblock it. You might lose connectivity to the router if you enable REP in a Telnet session that accesses the router through the same interface.
- You cannot run REP and STP or REP and Flex Links on the same segment or interface.
- If you connect an STP network to an REP segment, be sure that the connection is at the segment edge. An STP connection that is not at the edge might cause a bridging loop because STP does not run on REP segments. All the STP BPDUs are dropped at REP interfaces.
- You must configure all the trunk ports in a segment with the same set of allowed VLANs. If this is not done, misconfiguration occurs.
- If REP is enabled on two ports on a switch, both the ports must be either regular segment ports or edge ports. REP ports follow these rules:
 - There is no limit to the number of REP ports on a switch. However, only two ports on a switch can belong to the same REP segment.
 - If only one port on a switch is configured in a segment, the port should be an edge port.
 - If two ports on a switch belong to the same segment, they must both be edge ports, regular segment ports, or one regular port and one edge no-neighbor port. An edge port and regular segment port on a switch cannot belong to the same segment.
 - If two ports on a switch belong to the same segment, and one is configured as an edge port and one as a regular segment port (a misconfiguration), the edge port is treated as a regular segment port.
- REP interfaces come up in a blocked state and remain in a blocked state until they are safe to be unblocked. You must, therefore, be aware of the status of REP interfaces to avoid sudden connection losses.
- REP sends all the LSL PDUs in the untagged frames to the native VLAN. The BPA message sent to a Cisco multicast address is sent to the administration VLAN, which is VLAN 1 by default.
- You can configure the duration for which a REP interface remains up without receiving a hello from a neighbor. Use the **rep lsl-age-timer** value interface configuration command to set the time from 120 ms to 10000 ms. The LSL hello timer is then set to the age-timer value divided by 3. In normal operation, three LSL hellos are sent before the age timer on the peer switch expires and checks for hello messages.
 - EtherChannel port channel interfaces do not support LSL age-timer values less than 1000 ms. If you try to configure a value less than 1000 ms on a port channel, you receive an error message and the command is rejected.

- REP ports cannot be configured as one of the following port types:
 - Switched Port Analyzer (SPAN) destination port
 - Tunnel port
 - Access port
- REP is supported on EtherChannels, but not on an individual port that belongs to an EtherChannel.
- There can be a maximum of 64 REP segments per switch.

Configuring REP Administrative VLAN

To avoid the delay created by link-failure messages, and VLAN-blocking notifications during load balancing, REP floods packets to a regular multicast address at the hardware flood layer (HFL). These messages are flooded to the whole network, and not just the REP segment. You can control the flooding of these messages by configuring an administrative VLAN.

Follow these guidelines when configuring the REP administrative VLAN:

- If you do not configure an administrative VLAN, the default is VLAN 1.
- You can configure one admin VLAN on the switch for all segments.
- The administrative VLAN cannot be the RSPAN VLAN.

To configure the REP administrative VLAN, follow these steps, beginning in privileged EXEC mode:

SUMMARY STEPS

1. **configure terminal**
2. **rep admin vlan** *vlan-id*
3. **end**
4. **show interface** [*interface-id*] **rep detail**
5. **copy running-config startup config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 2	rep admin vlan <i>vlan-id</i> Example: Device(config)# rep admin vlan 2	Specifies the administrative VLAN. The range is from 2 to 4094. To set the admin VLAN to 1, which is the default, enter the no rep admin vlan global configuration command.

	Command or Action	Purpose
Step 3	end Example: Device(config)# end	Exits global configuration mode and returns to privileged EXEC mode.
Step 4	show interface [interface-id] rep detail Example: Device# show interface gigabitethernet1/1 rep detail	(Optional) Verifies the configuration on a REP interface.
Step 5	copy running-config startup config Example: Device# copy running-config startup config	(Optional) Saves your entries in the switch startup configuration file.

Configuring a REP Interface

To configure REP, enable REP on each segment interface and identify the segment ID. This task is mandatory, and must be done before other REP configurations. You must also configure a primary and secondary edge port on each segment. All the other steps are optional.

Follow these steps to enable and configure REP on an interface:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface interface-id**
4. **switchport mode trunk**
5. **rep segment segment-id [edge [no-neighbor] [primary]] [preferred]**
6. **rep stcn {interface interface id | segment id-list | stp}**
7. **rep block port {id port-id | neighbor-offset | preferred} vlan {vlan-list | all}**
8. **rep preempt delay seconds**
9. **rep lsl-age-timer value**
10. **end**
11. **show interface [interface-id] rep [detail]**
12. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.

	Command or Action	Purpose
Step 2	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode.
Step 3	interface <i>interface-id</i> Example: Device# <code>interface gigabitethernet1/1</code>	Specifies the interface, and enters interface configuration mode. The interface can be a physical Layer 2 interface or a port channel (logical interface).
Step 4	switchport mode trunk Example: Device# <code>switchport mode trunk</code>	Configures the interface as a Layer 2 trunk port.
Step 5	rep segment <i>segment-id</i> [edge [no-neighbor] [primary]] [preferred] Example: Device# <code>rep segment 1 edge no-neighbor primary</code>	<p>Enables REP on the interface and identifies a segment number. The segment ID range is from 1 to 1024.</p> <p>Note You must configure two edge ports, including one primary edge port, for each segment.</p> <p>These optional keywords are available:</p> <ul style="list-style-type: none"> • (Optional) edge—Configures the port as an edge port. Each segment has only two edge ports. Entering the keyword edge without the keyword primary configures the port as the secondary edge port. • (Optional) primary—Configures the port as the primary edge port, the port on which you can configure VLAN load balancing. • (Optional) no-neighbor—Configures a port with no external REP neighbors as an edge port. The port inherits all the properties of an edge port, and you can configure the properties the same way you would for an edge port. <p>Note Although each segment can have only one primary edge port, if you configure edge ports on two different switches and enter the keyword primary on both the switches, the configuration is valid. However, REP selects only one of these ports as the segment primary edge port. You can identify the primary edge port for a segment by entering the show rep topology privileged EXEC command.</p> <ul style="list-style-type: none"> • (Optional) preferred—Indicates that the port is the preferred alternate port or the preferred port for VLAN load balancing.

	Command or Action	Purpose
		<p>Note Configuring a port as preferred does not guarantee that it becomes the alternate port; it merely gives the port a slight edge over equal contenders. The alternate port is usually a previously failed port.</p>
<p>Step 6</p>	<p>rep stcn {interface <i>interface id</i> segment <i>id-list</i> stp}</p> <p>Example:</p> <pre>Device# rep stcn segment 25-50</pre>	<p>(Optional) Configures the edge port to send segment topology change notices (STCNs).</p> <ul style="list-style-type: none"> • interface <i>interface-id</i>—Designates a physical interface or port channel to receive STCNs. • segment <i>id-list</i>—Identifies one or more segments to receive STCNs. The range is from 1 to 1024. • stp—Sends STCNs to STP networks. <p>Note Spanning Tree (MST) mode is required on edge no-neighbor nodes when rep stcn stp command is configured for sending STCNs to STP networks.</p>
<p>Step 7</p>	<p>rep block port {id <i>port-id</i> <i>neighbor-offset</i> preferred} vlan {<i>vlan-list</i> all}</p> <p>Example:</p> <pre>Device# rep block port id 0009001818D68700 vlan 1-100</pre>	<p>(Optional) Configures VLAN load balancing on the primary edge port, identifies the REP alternate port in one of three ways (id port-id, <i>neighbor_offset</i>, preferred), and configures the VLANs to be blocked on the alternate port.</p> <ul style="list-style-type: none"> • id port-id—Identifies the alternate port by port ID. The port ID is automatically generated for each port in the segment. You can view interface port IDs by entering the show interface type number rep [detail] privileged EXEC command. • <i>neighbor_offset</i>—Number to identify the alternate port as a downstream neighbor from an edge port. The range is from -256 to 256, with negative numbers indicating the downstream neighbor from the secondary edge port. A value of 0 is invalid. Enter -1 to identify the secondary edge port as the alternate port. <p>Note Because you enter the rep block port command at the primary edge port (offset number 1), you cannot enter an offset value of 1 to identify an alternate port.</p> <ul style="list-style-type: none"> • preferred—Selects the regular segment port previously identified as the preferred alternate port for VLAN load balancing. • vlan <i>vlan-list</i>—Blocks one VLAN or a range of VLANs.

	Command or Action	Purpose
		<ul style="list-style-type: none"> • vlan all—Blocks all the VLANs. <p>Note Enter this command only on the REP primary edge port.</p>
Step 8	rep preempt delay <i>seconds</i> Example: Device# rep preempt delay 100	(Optional) Configures a preempt time delay. <ul style="list-style-type: none"> • Use this command if you want VLAN load balancing to be automatically triggered after a link failure and recovery. • The time delay range is between 15 to 300 seconds. The default is manual preemption with no time delay. <p>Note Enter this command only on the REP primary edge port.</p>
Step 9	rep lsl-age-timer <i>value</i> Example: Device# rep lsl-age-timer 2000	(Optional) Configures a time (in milliseconds) for which the REP interface remains up without receiving a hello from a neighbor. The range is from 120 to 10000 ms in 40-ms increments. The default is 5000 ms (5 seconds). <p>Note</p> <ul style="list-style-type: none"> • EtherChannel port channel interfaces do not support LSL age-timer values that are less than 1000 ms. • Both the ports on the link should have the same LSL age configured in order to avoid link flaps.
Step 10	end Example: Device(config)# end	Exits global configuration mode and returns to privileged EXEC mode.
Step 11	show interface [<i>interface-id</i>] rep [detail] Example: Device(config)# show interface gigabitethernet1/1 rep detail	(Optional) Displays the REP interface configuration.
Step 12	copy running-config startup-config Example: Device(config)# copy running-config startup-config	(Optional) Saves your entries in the router startup configuration file.

Setting Manual Preemption for VLAN Load Balancing

If you do not enter the **rep preempt delay seconds** interface configuration command on the primary edge port to configure a preemption time delay, the default is to manually trigger VLAN load balancing on the segment. Be sure that all the other segment configurations have been completed before manually preempting VLAN load balancing. When you enter the **rep preempt delay segment segment-id** command, a confirmation message is displayed before the command is executed because preemption might cause network disruption.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **rep preempt segment segment-id**
4. **show rep topology segment segment-id**
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	rep preempt segment segment-id Example: Device# rep preempt segment 100 The command will cause a momentary traffic disruption. Do you still want to continue? [confirm]	Manually triggers VLAN load balancing on the segment. You need to confirm the command before it is executed.
Step 4	show rep topology segment segment-id Example: Device# show rep topology segment 100	(Optional) Displays REP topology information.
Step 5	end Example: Device# end	Exits privileged EXEC mode.

Configuring SNMP Traps for REP

You can configure a router to send REP-specific traps to notify the Simple Network Management Protocol (SNMP) server of link-operational status changes and port role changes.

SUMMARY STEPS

1. **configure terminal**
2. **snmp mib rep trap-rate *value***
3. **end**
4. **show running-config**
5. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode.
Step 2	snmp mib rep trap-rate <i>value</i> Example: Device(config)# <code>snmp mib rep trap-rate 500</code>	Enables the switch to send REP traps, and sets the number of traps sent per second. <ul style="list-style-type: none"> • Enter the number of traps sent per second. The range is from 0 to 1000. The default is 0 (no limit is imposed; a trap is sent at every occurrence).
Step 3	end Example: Device(config)# <code>end</code>	Returns to privileged EXEC mode.
Step 4	show running-config Example: Device# <code>show running-config</code>	(Optional) Displays the running configuration, which can be used to verify the REP trap configuration.
Step 5	copy running-config startup-config Example: Device# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the switch startup configuration file.

Monitoring Resilient Ethernet Protocol Configurations

You can display the rep interface and rep topology details using the commands in this topic.

SUMMARY STEPS

1. **show interface** [*interface-id*] **rep** [**detail**]
2. **show rep topology** [*segment segment-id*] [**archive**] [**detail**]

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p>show interface [<i>interface-id</i>] rep [detail]</p> <p>Example:</p> <pre>Device# show interfaces TenGigabitEthernet4/1 rep detail TenGigabitEthernet4/1 REP enabled Segment-id: 3 (Primary Edge) PortID: 03010015FA66FF80 Preferred flag: No Operational Link Status: TWO_WAY Current Key: 02040015FA66FF804050 Port Role: Open Blocked VLAN: <empty> Admin-vlan: 1 Preempt Delay Timer: disabled Configured Load-balancing Block Port: none Configured Load-balancing Block VLAN: none STCN Propagate to: none LSL PDU rx: 999, tx: 652 HFL PDU rx: 0, tx: 0 BPA TLV rx: 500, tx: 4 BPA (STCN, LSL) TLV rx: 0, tx: 0 BPA (STCN, HFL) TLV rx: 0, tx: 0 EPA-ELECTION TLV rx: 6, tx: 5 EPA-COMMAND TLV rx: 0, tx: 0 EPA-INFO TLV rx: 135, tx: 136</pre>	<p>Displays REP configuration and status for an interface or for all the interfaces.</p> <ul style="list-style-type: none"> • (Optional) detail—Displays interface-specific REP information.
Step 2	<p>show rep topology [<i>segment segment-id</i>] [archive] [detail]</p> <p>Example:</p> <pre>Device# show rep topology REP Segment 1 BridgeName PortName Edge Role ----- 10.64.106.63 Te5/4 Pri Open 10.64.106.228 Te3/4 Open 10.64.106.228 Te3/3 Open 10.64.106.67 Te4/3 Open 10.64.106.67 Te4/4 Alt 10.64.106.63 Te4/4 Sec Open REP Segment 3 BridgeName PortName Edge Role ----- 10.64.106.63 Gi50/1 Pri Open SVT_3400_2 Gi0/3 Open SVT_3400_2 Gi0/4 Open 10.64.106.68 Gi40/2 Open</pre>	<p>Displays REP topology information for a segment or for all the segments, including the primary and secondary edge ports in the segment.</p> <ul style="list-style-type: none"> • (Optional) archive—Displays the last stable topology. <p>Note An archive topology is not retained when the switch reloads.</p> <ul style="list-style-type: none"> • (Optional) detail—Displays detailed archived information.

Command or Action	Purpose
10.64.106.68 Gi40/1 Open 10.64.106.63 Gi50/2 Sec Alt	

Configuration Examples for Resilient Ethernet Protocol

This section provides the following configuration examples:

Example: Configuring the REP Administrative VLAN

This example shows how to configure the administrative VLAN as VLAN 100, and verify the configuration by entering the **show interface rep detail** command on one of the REP interfaces:

```
Device# configure terminal
Device(config)# rep admin vlan 100
Device(config)# end
Device# show interface gigabitethernet1/1 rep detail

GigabitEthernet1/1 REP enabled
Segment-id: 2 (Edge)
PortID: 00010019E7144680
Preferred flag: No
Operational Link Status: TWO_WAY
Current Key: 0002001121A2D5800E4D
Port Role: Open
Blocked Vlan: <empty>
Admin-vlan: 100
Preempt Delay Timer: disabled
LSL Ageout Timer: 5000 ms
Configured Load-balancing Block Port: none
Configured Load-balancing Block VLAN: none
STCN Propagate to: none
LSL PDU rx: 3322, tx: 1722
HFL PDU rx: 32, tx: 5
BPA TLV rx: 16849, tx: 508
BPA (STCN, LSL) TLV rx: 0, tx: 0
BPA (STCN, HFL) TLV rx: 0, tx: 0
EPA-ELECTION TLV rx: 118, tx: 118
EPA-COMMAND TLV rx: 0, tx: 0
EPA-INFO TLV rx: 4214, tx: 4190
```

The following example shows how to create an administrative VLAN per segment. Here, VLAN 2 is configured as the administrative VLAN only for REP segment 2. All the remaining segments that are not configured have VLAN 1 as the administrative VLAN by default.

```
Device# configure terminal
Device(config)# rep admin vlan 2 segment 2
Device(config)# end
```

Example: Configuring a REP Interface

This example shows how to configure an interface as the primary edge port for segment 1, to send STCNs to segments 2 through 5, and to configure the alternate port as the port with port ID 0009001818D68700 to block

all the VLANs after a preemption delay of 60 seconds after a segment port failure and recovery. The interface is configured to remain up for 6000 ms without receiving a hello from a neighbor.

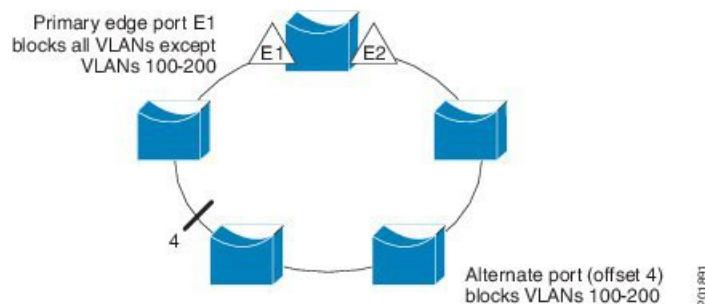
```
Switch# configure terminal
Switch (conf)# interface gigabitethernet1/1
Switch (conf-if)# rep segment 1 edge primary
Switch (conf-if)# rep stcn segment 2-5
Switch (conf-if)# rep block port 0009001818D68700 vlan all
Switch (conf-if)# rep preempt delay 60
Switch (conf-if)# rep lsl-age-timer 6000
Switch (conf-if)# end
```

This example shows how to configure the same configuration when the interface has no external REP neighbor:

```
Switch# configure terminal
Switch (conf)# interface gigabitethernet1/1
Switch (conf-if)# rep segment 1 edge no-neighbor primary
Switch (conf-if)# rep stcn segment 2-5
Switch (conf-if)# rep block port 0009001818D68700 vlan all
Switch (conf-if)# rep preempt delay 60
Switch (conf-if)# rep lsl-age-timer 6000
Switch (conf-if)# end
```

This example shows how to configure the VLAN blocking configuration shown in the Figure 5. The alternate port is the neighbor with neighbor offset number 4. After manual preemption, VLANs 100 to 200 are blocked at this port, and all the other VLANs are blocked at the primary edge port E1 (Gigabit Ethernet port 1/1).

Figure 59: Example of VLAN Blocking



```
Switch# configure terminal
Switch (conf)# interface gigabitethernet1/1
Switch (conf-if)# rep segment 1 edge primary
Switch (conf-if)# rep block port 4 vlan 100-200
Switch (conf-if)# end
```




CHAPTER 33

Configuring the PPPoE Intermediate Agent

- [Restrictions for PPPoE Intermediate Agent, on page 565](#)
- [Information about PPPoE Intermediate Agent, on page 565](#)
- [How to Configure PPPoE IA, on page 566](#)
- [Configuration Examples for PPPoE IA, on page 575](#)
- [Displaying Configuration Parameters, on page 577](#)
- [Clearing Packet Counters, on page 579](#)
- [Debugging PPPoE Intermediate Agent, on page 579](#)
- [Troubleshooting Tips, on page 580](#)
- [Feature Information for Configuring the PPPoE Intermediate Agent, on page 581](#)

Restrictions for PPPoE Intermediate Agent

PPPoE Intermediate Agent is not supported on routed interfaces.

Information about PPPoE Intermediate Agent

PPPoE Intermediate Agent (PPPoE IA) is placed between a subscriber and BRAS to help the service provider BRAS distinguish between end hosts connected over Ethernet to an access switch. On the access switch, PPPoE IA enables Subscriber Line Identification by appropriately tagging Ethernet frames of different users. (The tag contains specific information such as which subscriber is connected to the switch and VLAN.) PPPoE IA acts as mini security firewall between host and BRAS by intercepting all PPPoE Active Discovery (PAD) messages on a per-port per-VLAN basis. It provides specific security feature such as verifying the intercepted PAD message from untrusted port, performing per-port PAD message rate limiting, inserting and removing VSA Tags into and from PAD messages, respectively.

DSL Forum TR-101 [1] offers a means by which the PPPoE Discovery packets are tagged at the service provider's access switch with subscriber line specific information. The mechanism specifies using VSA of the PPPoE Discovery packets to add the line specific information at the switch. Even though you can perform Subscriber Line Identification (SLI) in another way (recreating virtual paths and circuits using stacked VLAN tags), DSL Forum 2004-071 [4] recommends the PPPoE Intermediate Agent mechanism. It cites lower provisioning costs and simpler co-ordination between OSS systems in charge of access switch and BRAS. PPPoE Intermediate Agent helps the service provider, BRAS, distinguish between end hosts connected over Ethernet to an access switch.

How to Configure PPPoE IA

Enabling PPPoE IA on a Switch



Note By default, PPPoE IA is disabled globally.

Follow these steps to enable or disable PPPoE IA globally on the switch:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **pppoe intermediate-agent**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	configure terminal Example: Device# config t	Enters the global configuration mode.
Step 3	pppoe intermediate-agent Example: Device(config)# pppoe intermediate-agent	Enables PPPoE IA globally on the switch.

Configuring the Access Node Identifier for PPPoE IA on a Switch

Follow these steps to set the Access Node Identifier of the switch.



Note By default, access-node-id is not set.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **pppoe intermediate-agent** *format-type access-node-id string word*

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	configure terminal Example: Device# config terminal	Enters the global configuration mode.
Step 3	pppoe intermediate-agent format-type access-node-id string word Example: Device(config)# pppoe intermediate-agent format-type access-node-id string abcd	Sets the access node identifier. • access-node-identifier string word – ASCII string literal value for the access-node-identifier.

Configuring the Identifier String, Option, and Delimiter for PPPoE IA on a Switch

This functionality overrides the default automatic generation of circuit-id by the system.

The options available are sp, sv, pv and spv denoting slot:port, slot-vlan, port-vlan, and slot-port-vlan combinations, respectively. Valid delimiters are # . , ; / space.

The no form of this command without WORD, options, and delimiters, reverts to the default automatic generation of circuit-id.

This command does not affect the circuit ID configured explicitly per-interface or per-interface per-VLAN with the **pppoe intermediate-agent format-type circuit-id**.

Follow these steps to set an identifier string word with option spv delimited by “.”

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **pppoe intermediate-agent format-type identifier-string string wordoption {sp | sv | pv | spv} delimiter {, | . | ; | / | #}**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	configure terminal Example: Device# config t	Enters the global configuration mode.

	Command or Action	Purpose
Step 3	<p>pppoe intermediate-agent format-type identifier-string string <i>word</i> option {sp sv pv spv} delimiter {, . ; / #}</p> <p>Example:</p> <pre>Device(config)# pppoe intermediate-agent format-type identifier-string string word option spv delimiter :</pre>	<p>Sets the identifier string.</p> <ul style="list-style-type: none"> option {sp sv pv spv} – sp = slot + port, sv = slot + vlan, pv = port + vlan, spv = slot +port+vlan. delimiter {, . ; / #} – Delimiter between slot/port/vlan portions of 'option'.

Configuring the Generic Error Message for PPPoE IA on a Switch

Follow these steps to to configure a generic message of **packet_length>1484**:

SUMMARY STEPS

1. enable
2. configure terminal
3. pppoe intermediate-agentformat-type generic-error-message string*string*

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p>enable</p> <p>Example:</p> <pre>Device> enable</pre>	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	<p>configure terminal</p> <p>Example:</p> <pre>Device# config terminal</pre>	Enters the global configuration mode.
Step 3	<p>pppoe intermediate-agentformat-type generic-error-message string<i>string</i></p> <p>Example:</p> <pre>Device(config)# pppoe intermediate-agent format-type generic-error-message string packet_length>1484</pre>	<p>Sets the generic error message.</p> <ul style="list-style-type: none"> generic-error-message string <i>string</i> – ASCII string literal value for the generic-error-message.

Enabling PPPoE IA on an Interface

Follow these steps to enable PPPoE IA on FastEthernet 3/1:

Before you begin

Note Enabling PPPoE IA on an interface does not ensure that incoming packets are tagged. For this to happen PPPoE IA must be enabled globally, and at least one interface that connects the switch to PPPoE server has a trusted PPPoE IA setting. Refer to the following section for details.

This functionality enables the PPPoE IA feature on an interface. The `pppoe intermediate-agent` command has an effect only if the PPPoE IA feature was enabled globally with this command. (You need to enable globally to activate PPPoE IA static ACL and on an interface for PPPoE IA processing of PPPoE discovery packets received on that interface.)

This setting applies to all frames passing through this interface, regardless of the VLAN they belong to. By default the PPPoE IA feature is disabled on all interfaces. You need to run this command on every interface that requires this feature.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface***interface-id*
4. **pppoe intermediate-agent**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	configure terminal Example: Device# config terminal	Enters the global configuration mode.
Step 3	interface <i>interface-id</i> Example: Device(config)interface FastEthernet 3/1	Enter interface configuration mode and the physical interface identification.
Step 4	pppoe intermediate-agent Example: Device(config-if)pppoe intermediate-agent	Enables PPPoE IA on the interface.

Configuring the PPPoE IA Trust Setting on an Interface



Note Interfaces that connect the switch to PPPoE server are configured as trusted. Interfaces that connect the switch to users (PPPoE clients) are untrusted.

Follow these steps to to set FastEthernet interface 3/2 as trusted:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface***interface-id*
4. **pppoe intermediate-agent trust**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	configure terminal Example: Device# config terminal	Enters the global configuration mode.
Step 3	interface <i>interface-id</i> Example: Device(config)interface FastEthernet 3/2	Enter interface configuration mode and the physical interface identification.
Step 4	pppoe intermediate-agent trust Example: Device(config-if)pppoe intermediate-agent trust	Sets the trust configuration of an interface.

Configuring PPPoE Intermediate Agent Rate Limiting Setting on an Interface



Note The parameter for rate limiting is the number of packets per second. If the incoming packet rate exceeds this value, the port shuts down.

Follow these steps to to set a rate limit on an interface :

SUMMARY STEPS

1. **enable**

2. **configure terminal**
3. **interface***interface-id*
4. **pppoe intermediate-agent limit rate***number*

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	configure terminal Example: Device# config terminal	Enters the global configuration mode.
Step 3	interface <i>interface-id</i> Example: Device(config)interface FastEthernet 3/1	Enters interface configuration mode and the physical interface identification.
Step 4	pppoe intermediate-agent limit rate <i>number</i> Example: Device(config-if)pppoe intermediate-agent limit rate 30	Limits the rate of the PPPoE Discovery packets arriving on an interface.

Configuring PPPoE IA Vendor-tag Stripping on an Interface



Note Generally, you would configure vendor-tag stripping on an interfaces connected to the PPPoE server. If you configure stripping, incoming packets are stripped of their VSAs (which carry subscriber and line identification information). For this to happen, the PPPoE Intermediate agent must be enabled to make the pppoe intermediate-agent vendor-tag strip command effective, and the interface must be set to trust. In isolation, the command has no effect.



Note BRAS automatically strips the vendor-specific tag off of the PPPoE discovery packets before sending them downstream to the access switch. To operate with older BRAS which does not possess this capability, use the pppoe intermediate-agent vendor-tag strip command on the interface connecting the access switch to BRAS.

Follow these steps to enable vendor-tag stripping :

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface***interface-id*

4. pppoe intermediate-agent vendor-tag strip

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	configure terminal Example: Device# config terminal	Enters the global configuration mode.
Step 3	interface <i>interface-id</i> Example: Device(config)interface FastEthernet 3/2	Enters interface configuration mode and the physical interface identification.
Step 4	pppoe intermediate-agent vendor-tag strip Example: Device(config-if)pppoe intermediate-agent vendor-tag strip	Enables vendor-tag stripping on PPPoE Discovery packets from PPPoE Server (or BRAS).

Configuring PPPoE Intermediate Agent Circuit-ID and Remote-ID on an Interface

The **[no] pppoe intermediate-agent format-type circuit-id** command sets the circuit ID on an interface and overrides the automatic generation of circuit ID by the switch. Without this command, one default tag (for example, Ethernet x/y:z on the PPPoE to which the user is connected) inserted by an intermediate-agent.

The **[no] pppoe intermediate-agent format-type remote-id** command sets the remote ID on an interface.

This functionality causes tagging of PADI, PADR, and PADT packets (belonging to PPPoE Discovery stage) received on this physical interface with circuit ID or remote ID. This happens regardless of their VLAN if PPPoE IA is not enabled for that VLAN.

You should use remote ID instead of circuit ID for subscriber line identification. You should configure this setting on every interface where you enabled PPPoE IA because it is not set by default. The default value for remote-id is the switch MAC address (for all physical interfaces).

Follow these steps to configure the circuit ID as root and the remote ID as granite:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface***interface-id*
4. **pppoe intermediate-agent format-type {circuit-id | remote-id} string***string*

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	configure terminal Example: Device# config terminal	Enters the global configuration mode.
Step 3	interface <i>interface-id</i> Example: Device(config)interface FastEthernet 3/1	Enters interface configuration mode and the physical interface identification.
Step 4	pppoe intermediate-agent format-type {circuit-id remote-id} string <i>string</i> Example: Device(config-if)pppoe intermediate-agent format-type circuit-id string root	Sets circuit-id or remote-id for an interface. <ul style="list-style-type: none"> • circuit-id string string – ASCII string literal value for circuit-id. • remote-id string string – ASCII string literal value for remote-id.

Enabling PPPoE IA for a Specific VLAN on an Interface



Note The **pppoe intermediate-agent** command in the vlan-range mode is not dependent on the same command in interface mode. The **pppoe intermediate-agent** command will take effect independently of the command in the interface mode. To make this happen, PPPoE IA must be enabled globally and at least one interface is connected to the PPPoE server.

Follow these steps to to enable PPPoE IA on a specific VLAN:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface***interface-id*
4. **vlan-range** {*vlan-id* |*vlan-list*|*vlan-range*}
5. **pppoe intermediate-agent**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example:	Enables privileged EXEC mode. Enter your password if prompted.

	Command or Action	Purpose
	Device> enable	
Step 2	configure terminal Example: Device# config terminal	Enters the global configuration mode.
Step 3	interface <i>interface-id</i> Example: Device(config) interface FastEthernet 3/1	Enters interface configuration mode and the physical interface identification.
Step 4	vlan-range { <i>vlan-id</i> <i>vlan-list</i> <i>vlan-range</i> } Example: Device(config-if) vlan-range 5	Enters the vlan-range mode.
Step 5	pppoe intermediate-agent Example: Device(config-if-vlan-range) pppoe intermediate-agent	Enables PPPoE IA on the specified interfaces.

Configuring PPPoE IA Circuit-ID and Remote-ID for a VLAN on an Interface



Note The circuit-id and remote-id configurations in vlan-range mode are affected only if PPPoE IA is enabled globally and in vlan-range mode.



Note The vlan-range mode commands configure PPPoE IA for either a specific VLAN, multiple VLANs, or VLAN range, depending on what you specify in the syntax.

In this section you set the circuit ID and remote ID for a specific VLAN on an interface. The command overrides the circuit ID and remote ID specified for this physical interface and the switch uses the WORD value to tag packets received on this VLAN. This parameter is unset by default.

The default value of remote-id is the switch MAC address (for all VLANs). You would set this parameter to encode subscriber-specific information.

Follow these steps to set the circuit-id and the remote-id :

SUMMARY STEPS

1. **interface***interface-id*
2. **vlan-range***vlan-range*
3. **pppoe intermediate-agent**
4. **pppoe intermediate-agent format-type** {*circuit-id* | *remote-id*} *stringstring*

DETAILED STEPS

	Command or Action	Purpose
Step 1	interface <i>interface-id</i> Example: Device(config)# int g3/7	Enters interface configuration mode and the physical interface identification.
Step 2	vlan-range <i>vlan-range</i> Example: Device(config-if)# vlan-range5	Enters the vlan-range mode.
Step 3	pppoe intermediate-agent Example: Device(config-if)# pppoe intermediate-agent	Enables PPPoE IA on the specified interfaces.
Step 4	pppoe intermediate-agent format-type {circuit-id remote-id} string <i>string</i> Example: Device(config-if) pppoe intermediate-agent format-type circuit-id string root	Sets circuit-id or remote-id for an interface. <ul style="list-style-type: none"> • circuit-id string string – ASCII string literal value for circuit-id. • remote-id string string – ASCII string literal value for remote-id.

Configuration Examples for PPPoE IA

Example: Enabling PPPoE Intermediate Agent on a Switch

This examples shows how to enable or disable PPPoE IA globally on the switch

```
Device> enable
Device# configure terminal
Device(config)# pppoe intermediate-agent
```

Example: Configuring the Access Node Identifier for PPPoE IA on a Switch

This example shows how to to set an access node identifier of abcd:

```
Device> enable
Device# configure terminal
Device(config)#pppoe intermediate-agent format-type access-node-id string abcd
```

Example: Configuring the Identifier String, Option, and Delimiter for PPPoE IA on a Switch

This example shows how to set an identifier string word with option spv delimited by “:”:

```
Device> enable
Device# configure terminal
```

```
Device(config)#pppoe intermediate-agent format-type
  identifier-string string word
option spv delimiter :
```

Example: Configuring the Generic Error Message for PPPoE IA on a Switch

This example shows how to configure a generic message of packet_length>1484:

```
Device> enable
Device# configure terminal
Device(config)#pppoe intermediate-agent format-type generic-error-message string
packet_length>1484
```

Example: Enabling PPPoE IA on an Interface

This example shows how to enable PPPoE IA on FastEthernet 3/1:

```
Device> enable
Device# configure terminal
Device(config) interface FastEthernet 3/1
Device(config-if)pppoe intermediate-agent
```

Example: Configuring the PPPoE Intermediate Agent Trust Setting on an Interface

The following example shows how to set FastEthernet interface 3/2 as trusted:

```
Device> enable
Device# configure terminal
Device(config)interface FastEthernet 3/2
Device(config-if)pppoe intermediate-agent trust
```

Example: Configuring PPPoE Intermediate Agent Rate Limiting Setting on an Interface

This example shows how to set a rate limit of 30 at FastEthernet 3/1:

```
Device> enable
Device# configure terminal
Device(config) interface FastEthernet 3/1
Device(config-if)pppoe intermediate-agent limit rate 30
```

Example: Configuring PPPoE IA Vendor-tag Stripping on an Interface

The following example shows how to to enable stripping on FastEthernet 3/2:

```
Device> enable
Device# configure terminal
Device(config)interface FastEthernet 3/2
Device(config-if)pppoe intermediate-agent vendor-tag strip
```

Example: Configuring PPPoE IA Circuit-ID and Remote-ID on an Interface

The following example shows how to configure the circuit ID as root and the remote ID as granite:

```
Device> enable
Device# configure terminal
Device(config) interface FastEthernet 3/1
Device(config-if)pppoe intermediate-agent format-type circuit-id string root
Device(config-if)pppoe intermediate-agent format-type remote-id string granite
```

Example: Enabling PPPoE IA for a Specific VLAN on an Interface

The following example shows how to enable PPPOE IA on a specific VLAN:

```
Switch# configure terminal
Switch(config)# interface FastEthernet 3/1
Switch(config-if)# vlan-range 5
Switch(config-if-vlan-range)# pppoe intermediate-agent
```

The following examples shows how to enable PPPoE IA on a comma-separated VLAN list

```
Switch# configure terminal
Switch(config)# interface FastEthernet 3/1
Switch(config-if)# vlan-range 5,6
Switch(config-if-vlan-range)# pppoe intermediate-agent
```

The following example shows how to enable PPPoE IA on a VLAN-range such as “x-y.”

```
Switch# configure terminal
Switch(config)# interface FastEthernet 3/1
Switch(config-if)# vlan-range 5-9
Switch(config-if-vlan-range)# pppoe intermediate-agent
```

Example: Configuring PPPoE IA Circuit-ID and Remote-ID for a VLAN on an Interface

The following example shows how to set the circuit-id to aaa and the remote-id as ccc on interface g3/7:

```
Switch(config)# int g3/7
Switch(config-if)# vlan-range 5
Switch(config-if)# pppoe intermediate-agent
Switch(config-if-vlan-range)# pppoe intermediate-agent format-type circuit-id string aaa
Switch(config-if-vlan-range)# pppoe intermediate-agent format-type remote-id string ccc
```

Displaying Configuration Parameters

The **show pppoe intermediate-agent [info | statistics][interface {interface}]** command displays the various configuration parameters, statistics, and counters stored for PPPoE.

The **info** keyword appears if the PPPoE Intermediate Agent is enabled globally on an interface or on a VLAN (in an interface). It also informs you about the access node ID and generic error message of the switch, as well as the identifier string options, delimiter values configured globally, global circuit id and remote id configuration by using the following command:

```
Switch(config)# pppoe intermediate-agent format-type ?
access-node-id      Access Node Identifier
circuit-id          Circuit Id
```

```
generic-error-message  Generic Error Message
identifier-string      Identifier String
remote-id              Remote Id
```

The **info** keyword also displays the circuit ID, remote ID, trust and rate limit configurations, and vendor tag strip setting for all interfaces and for all VLANs pertaining to those interfaces. If any of these parameters are not set, they are not displayed.

The **statistics** option displays the number of PADI/PADR/PADT packets received, and the time the last packet was received on all interfaces and on all VLANs pertaining to those interfaces.

If **interface** is specified, information or statistics applicable only to that physical interface and pertaining VLANs is displayed.

Although PPoE IA is supported on PVLANS, be aware that no PVLAN association (primary and secondary VLAN mapping) information is displayed.

The PPPoE IA show commands such as **show pppoe intermediate-agent info**, **show pppoe intermediate-agent info interface g3/7**, or **show pppoe intermediate-agent statistics** do not provide information about private VLAN association (primary and secondary VLAN mapping).

However, they do provide information about VLANs regardless of private or normal VLANs, as the following example illustrate:

```
Switch# show pppoe intermediate-agent info
Switch PPPOE Intermediate-Agent is enabled
PPPOE Intermediate-Agent trust/rate is configured on the following Interfaces:
```

Interface	IA	Trusted	Vsa Strip	Rate limit (pps)
GigabitEthernet3/4	no	yes	yes	unlimited PPPOE
Intermediate-Agent is configured on following VLANs: 2-3				
GigabitEthernet3/7	no	no	no	unlimited OE Intermediate-Agent
is configured on following VLANs: -3				

```
Switch# show pppoe intermediate-agent info interface g3/7
Interface      IA      Trusted  Vsa Strip  Rate limit (pps)
-----
GigabitEthernet3/7  yes    no       no         unlimited oE Intermediate-Agent
is configured on following VLANs:
-3
```

```
Switch# show pppoe intermediate-agent statistics
PPPOE IA Per-Port Statistics
-----

Interface : GigabitEthernet3/7 Packets received
All = 0
PADI = 0 PADO = 0
PADR = 0 PADS = 0
PADT = 0
Packets dropped:
Rate-limit exceeded = 0
Server responses from untrusted ports = 0
Client requests towards untrusted ports = 0
Malformed PPPoE Discovery packets = 0
```

The following statistics will be displayed when PPPoE IA feature is enabled on every VLAN interface and the PAD packet counters have a non-zero value.

```
switch# sh run int gi2/0/1
Building configuration...
Current configuration : 135 bytes
!
interface GigabitEthernet2/0/1
switchport mode trunk
pppoe intermediate-agent
vlan-range 200-201
pppoe intermediate-agent
end

Switch# show pppoe intermediate-agent statistics interface gi2/0/3
Interface: GigabitEthernet2/0/3
Packets received
All = 0
PADI = 0 PADO = 0
PADR = 0 PADS = 0
PADT = 0
Packets dropped:
Rate-limit exceeded = 0
Server responses from untrusted ports = 0
Client requests towards untrusted ports = 0
Malformed PPPoE Discovery packets = 0

Switch# show pppoe intermediate-agent statistics interface gi2/0/3
Interface: GigabitEthernet2/0/1
Packets received
All = 50
PADI = 20 PADO = 0
PADR = 20 PADS = 0
PADT = 10
Packets dropped: Rate-limit exceeded = 0
Server responses from untrusted ports = 0
Client requests towards untrusted ports = 0
Malformed PPPoE Discovery packets = 0
Vlan 200: Packets received PADI = 2 PADO = 0 PADR = 2 PADS = 0 PADT = 1
Vlan 201: Packets received PADI = 2 PADO = 0 PADR = 2 PADS = 0 PADT = 1
```

Clearing Packet Counters

This section illustrates how to clear packet counters on all interfaces (per-port and per-port-per-VLAN).

The following example illustrates how to do this:

```
Switch# clear pppoe intermediate-agent statistics
Issuing of the above command clears the counters for all PPPoE discovery packets
(PADI, PADO, PADR, PADS, PADT) received on DUT.
```

Debugging PPPoE Intermediate Agent

The **debug pppoe intermediate-agent [packet | event | all]** command enables you to display useful PPPoE information that assists in debugging. This command is disabled by default.

The **packet** option of the command displays the contents of a packet received in the software: source and destination MAC address of Ethernet frame, code, version and type of PPPoE Discovery packet and a list of TAGs present.

The **event** option of the command echoes important messages (interface state change to errdisabled due to PPPoE discovery packets entering at a rate exceeding the configured limit). It is the only event shown by the **debug pppoe intermediate-agent event** command.

The **all** option enables both package and event options.

The following example illustrates how to enter the debug command with the packet option:

```
Switch# debug pppoe intermediate-agent packet
PPPOE IA Packet debugging is on
*Sep 2 06:12:56.133: PPPOE_IA: Process new PPPoE packet, Message type: PADI, input interface:

Gi3/7, vlan : 2 MAC da: ffff.ffff.ffff, MAC sa: aabb.cc00.0000
*Sep 2 06:12:56.137: PPPOE_IA: received new PPPoE packet from inputinterface
(GigabitEthernet3/4)
*Sep 2 06:12:56.137: PPPOE_IA: received new PPPoE packet from inputinterface
(GigabitEthernet3/8)
*Sep 2 06:12:56.137: PPPOE_IA: Process new PPPoE packet, Message type: PADO, input interface:

Gi3/4, vlan : 2 MAC da: aabb.cc00.0000, MAC sa: 001d.e64c.6512
*Sep 2 06:12:56.137: PPPOE_IA: Process new PPPoE packet, Message type: PADO, input interface:

Gi3/8, vlan : 2 MAC da: aabb.cc00.0000, MAC sa: aabb.cc80.0000
*Sep 2 06:12:56.137: PPPOE_IA: received new PPPoE packet from inputinterface
(GigabitEthernet3/7)
*Sep 2 06:12:56.137: PPPOE_IA: Process new PPPoE packet, Message type: PADR, input interface:

Gi3/7, vlan : 2 MAC da: 001d.e64c.6512, MAC sa: aabb.cc00.0000
*Sep 2 06:12:56.145: PPPOE_IA: received new PPPoE packet from inputinterface
(GigabitEthernet3/4)
*Sep 2 06:12:56.145: PPPOE_IA: Process new PPPoE packet, Message type: PAD      ut interface:

Gi3/4, vlan : 2 MAC da: aabb.cc00.0000, MAC sa: 001d.e64c.6512
```

The following example illustrates how to enter the debug command with the event option:

```
Switch# PPPOE I
*Jul 30 19:00:10.254: %PPPOE_IA-4-PPPOE_IA_ERRDISABLE_WARNING: PPPOE IA received 5 PPPOE
packets on interface Gi3/7
*Jul 30 19:00:10.254: %PPPOE_IA-4-PPPOE_IA_RATE_LIMIT_EXCEEDED: The interface Gi3/7 is
receiving more than the threshold set
*Jul 30 19:00:10.394: %PM-4-ERR_DISABLE:                               detected on
Gi3/7, putting Gi3/7 in err-disable stat
```

Troubleshooting Tips

When the **radius-server attribute 31 remote-id** global configuration command is entered in the PPPoE Agent Remote-ID Tag and DSL Line Characteristics feature configuration on the BRAS, the **debug radius** privileged EXEC command can be used to generate a report that includes information about the incoming access interface, where discovery frames are received, and about the session being established in PPPoE extended NAS-Port format (format d)

Feature Information for Configuring the PPPoE Intermediate Agent

Table 60: Feature Information for Configuring the PPPoE Intermediate Agent

Feature Name	Releases	Feature Information
PPPoE Intermediate Agent	Cisco IOS XE 15.2(6)E2	Supports Point-to-point protocol over Ethernet intermediate agent (PPPoE IA) which is placed between a subscriber and broadband remote access server (BRAS). PPPoE IA helps the service provider BRAS to distinguish between end hosts connected over Ethernet to an access switch.



PART VI

Network Management

- [Configuring Cisco IOS Configuration Engine, on page 585](#)
- [Configuring the Cisco Discovery Protocol, on page 605](#)
- [Configuring Simple Network Management Protocol, on page 617](#)
- [Configuring SPAN and RSPAN, on page 641](#)



CHAPTER 34

Configuring Cisco IOS Configuration Engine

- [Prerequisites for Configuring the Configuration Engine, on page 585](#)
- [Restrictions for Configuring the Configuration Engine, on page 585](#)
- [Information About Configuring the Configuration Engine, on page 586](#)
- [How to Configure the Configuration Engine, on page 591](#)
- [Monitoring CNS Configurations, on page 603](#)
- [Additional References, on page 603](#)
- [Feature History and Information for the Configuration Engine, on page 604](#)

Prerequisites for Configuring the Configuration Engine

- Obtain the name of the configuration engine instance to which you are connecting.
- Because the CNS uses both the event bus and the configuration server to provide configurations to devices, you must define both ConfigID and Device ID for each configured device.
- All devices configured with the **cns config partial** global configuration command must access the event bus. The DeviceID, as originated on the device, must match the DeviceID of the corresponding device definition in the Cisco Configuration Engine. You must know the hostname of the event bus to which you are connecting.

Restrictions for Configuring the Configuration Engine

- Within the scope of a single instance of the configuration server, no two configured devices can share the same value for ConfigID.
- Within the scope of a single instance of the event bus, no two configured devices can share the same value for DeviceID.

Information About Configuring the Configuration Engine

Cisco Configuration Engine Software

The Cisco Configuration Engine is network management utility software that acts as a configuration service for automating the deployment and management of network devices and services. Each Cisco Configuration Engine manages a group of Cisco devices (devices and routers) and the services that they deliver, storing their configurations and delivering them as needed. The Cisco Configuration Engine automates initial configurations and configuration updates by generating device-specific configuration changes, sending them to the device, executing the configuration change, and logging the results.

The Cisco Configuration Engine supports standalone and server modes and has these Cisco Networking Services (CNS) components:

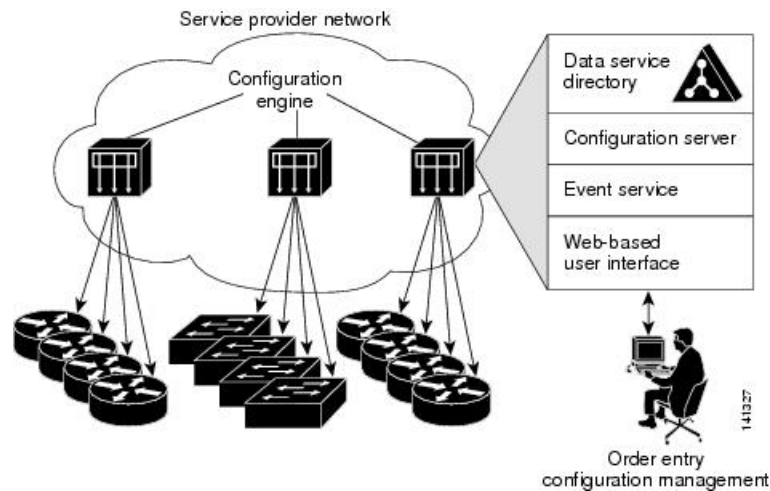
- Configuration service:
 - Web server
 - File manager
 - Namespace mapping server
- Event service (event gateway)
- Data service directory (data models and schema)



Note Support for Cisco Configuration Engine will be deprecated in future releases. Use the configuration described in [Cisco Plug and Play Feature Guide](#).

In standalone mode, the Cisco Configuration Engine supports an embedded directory service. In this mode, no external directory or other data store is required. In server mode, the Cisco Configuration Engine supports the use of a user-defined external directory.

Figure 60: Cisco Configuration Engine Architectural Overview



Configuration Service

The Configuration Service is the core component of the Cisco Configuration Engine. It consists of a Configuration Server that works with Cisco IOS CNS agents on the device. The Configuration Service delivers device and service configurations to the device for initial configuration and mass reconfiguration by logical groups. Devices receive their initial configuration from the Configuration Service when they start up on the network for the first time.

The Configuration Service uses the CNS Event Service to send and receive configuration change events and to send success and failure notifications.

The Configuration Server is a web server that uses configuration templates and the device-specific configuration information stored in the embedded (standalone mode) or remote (server mode) directory.

Configuration templates are text files containing static configuration information in the form of CLI commands. In the templates, variables are specified by using Lightweight Directory Access Protocol (LDAP) URLs that reference the device-specific configuration information stored in a directory.

The Cisco IOS agent can perform a syntax check on received configuration files and publish events to show the success or failure of the syntax check. The configuration agent can either apply configurations immediately or delay the application until receipt of a synchronization event from the configuration server.

Event Service

The Cisco Configuration Engine uses the Event Service for receipt and generation of configuration events. The Event Service consists of an event agent and an event gateway. The event agent is on the device and facilitates the communication between the device and the event gateway on the Cisco Configuration Engine.

The Event Service is a highly capable publish-and-subscribe communication method. The Event Service uses subject-based addressing to send messages to their destinations. Subject-based addressing conventions define a simple, uniform namespace for messages and their destinations.

NameSpace Mapper

The Cisco Configuration Engine includes the NameSpace Mapper (NSM) that provides a lookup service for managing logical groups of devices based on application, device or group ID, and event.

Cisco IOS devices recognize only event subject-names that match those configured in Cisco IOS software; for example, `cisco.cns.config.load`. You can use the namespace mapping service to designate events by using any desired naming convention. When you have populated your data store with your subject names, NSM changes your event subject-name strings to those known by Cisco IOS.

For a subscriber, when given a unique device ID and event, the namespace mapping service returns a set of events to which to subscribe. Similarly, for a publisher, when given a unique group ID, device ID, and event, the mapping service returns a set of events on which to publish.

Cisco Networking Services IDs and Device Hostnames

The Cisco Configuration Engine assumes that a unique identifier is associated with each configured device. This unique identifier can take on multiple synonyms, where each synonym is unique within a particular namespace. The event service uses namespace content for subject-based addressing of messages.

The Cisco Configuration Engine intersects two namespaces, one for the event bus and the other for the configuration server. Within the scope of the configuration server namespace, the term *ConfigID* is the unique identifier for a device. Within the scope of the event bus namespace, the term *DeviceID* is the CNS unique identifier for a device.

ConfigID

Each configured device has a unique ConfigID, which serves as the key into the Cisco Configuration Engine directory for the corresponding set of device CLI attributes. The ConfigID defined on the device must match the ConfigID for the corresponding device definition on the Cisco Configuration Engine.

The ConfigID is fixed at startup time and cannot be changed until the device restarts, even if the device hostname is reconfigured.

DeviceID

Each configured device participating on the event bus has a unique DeviceID, which is analogous to the device source address so that the device can be targeted as a specific destination on the bus.

The origin of the DeviceID is defined by the Cisco IOS hostname of the device. However, the DeviceID variable and its usage reside within the event gateway adjacent to the device.

The logical Cisco IOS termination point on the event bus is embedded in the event gateway, which in turn functions as a proxy on behalf of the device. The event gateway represents the device and its corresponding DeviceID to the event bus.

The device declares its hostname to the event gateway immediately after the successful connection to the event gateway. The event gateway couples the DeviceID value to the Cisco IOS hostname each time this connection is established. The event gateway retains this DeviceID value for the duration of its connection to the device.

Hostname and DeviceID

The DeviceID is fixed at the time of the connection to the event gateway and does not change even when the device hostname is reconfigured.

When changing the device hostname on the device, the only way to refresh the DeviceID is to break the connection between the device and the event gateway. For instructions on refreshing DeviceIDs, see "Related Topics."

When the connection is reestablished, the device sends its modified hostname to the event gateway. The event gateway redefines the DeviceID to the new value.



Caution When using the Cisco Configuration Engine user interface, you must first set the DeviceID field to the hostname value that the device acquires *after*, not *before*, and you must reinitialize the configuration for your Cisco IOS CNS agent. Otherwise, subsequent partial configuration command operations may malfunction.

Hostname, DeviceID, and ConfigID

In standalone mode, when a hostname value is set for a device, the configuration server uses the hostname as the DeviceID when an event is sent on hostname. If the hostname has not been set, the event is sent on the `cn=<value>` of the device.

In server mode, the hostname is not used. In this mode, the unique DeviceID attribute is always used for sending an event on the bus. If this attribute is not set, you cannot update the device.

These and other associated attributes (tag value pairs) are set when you run **Setup** on the Cisco Configuration Engine.

Cisco IOS CNS Agents

The CNS event agent feature allows the device to publish and subscribe to events on the event bus and works with the Cisco IOS CNS agent. These agents, embedded in the device Cisco IOS software, allow the device to be connected and automatically configured.

Initial Configuration

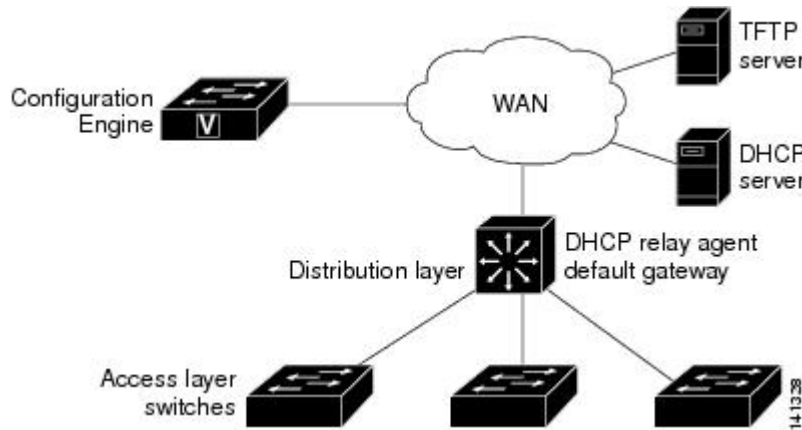
When the device first comes up, it attempts to get an IP address by broadcasting a Dynamic Host Configuration Protocol (DHCP) request on the network. Assuming there is no DHCP server on the subnet, the distribution device acts as a DHCP relay agent and forwards the request to the DHCP server. Upon receiving the request, the DHCP server assigns an IP address to the new device and includes the Trivial File Transfer Protocol (TFTP) server Internet Protocol (IP) address, the path to the bootstrap configuration file, and the default gateway IP address in a unicast reply to the DHCP relay agent. The DHCP relay agent forwards the reply to the device.

The device automatically configures the assigned IP address on interface VLAN 1 (the default) and downloads the bootstrap configuration file from the TFTP server. Upon successful download of the bootstrap configuration file, the device loads the file in its running configuration.

The Cisco IOS CNS agents initiate communication with the Configuration Engine by using the appropriate ConfigID and EventID. The Configuration Engine maps the Config ID to a template and downloads the full configuration file to the device.

The following figure shows a sample network configuration for retrieving the initial bootstrap configuration file by using DHCP-based autoconfiguration.

Figure 61: Initial Configuration



Incremental (Partial) Configuration

After the network is running, new services can be added by using the Cisco IOS CNS agent. Incremental (partial) configurations can be sent to the device. The actual configuration can be sent as an event payload by way of the event gateway (push operation) or as a signal event that triggers the device to initiate a pull operation.

The device can check the syntax of the configuration before applying it. If the syntax is correct, the device applies the incremental configuration and publishes an event that signals success to the configuration server. If the device does not apply the incremental configuration, it publishes an event showing an error status. When the device has applied the incremental configuration, it can write it to nonvolatile random-access memory (NVRAM) or wait until signaled to do so.

Synchronized Configuration

When the device receives a configuration, it can defer application of the configuration upon receipt of a write-signal event. The write-signal event tells the device not to save the updated configuration into its NVRAM. The device uses the updated configuration as its running configuration. This ensures that the device configuration is synchronized with other network activities before saving the configuration in NVRAM for use at the next reboot.

Automated CNS Configuration

To enable automated CNS configuration of the device, you must first complete the prerequisites listed in this topic. When you complete them, power on the device. At the **setup** prompt, do nothing; the device begins the initial configuration. When the full configuration file is loaded on your device, you do not need to do anything else.

For more information on what happens during initial configuration, see "Related Topics."

Table 61: Prerequisites for Enabling Automatic Configuration

Device	Required Configuration
Access device	Factory default (no configuration file)

Device	Required Configuration
Distribution device	<ul style="list-style-type: none"> • IP helper address • Enable DHCP relay agent² • IP routing (if used as default gateway)
DHCP server	<ul style="list-style-type: none"> • IP address assignment • TFTP server IP address • Path to bootstrap configuration file on the TFTP server • Default gateway IP address
TFTP server	<ul style="list-style-type: none"> • A bootstrap configuration file that includes the CNS configuration commands that enable the device to communicate with the Configuration Engine • The device configured to use either the device MAC address or the serial number (instead of the default hostname) to generate the ConfigID and EventID • The CNS event agent configured to push the configuration file to the device
CNS Configuration Engine	One or more templates for each type of device, with the ConfigID of the device mapped to the template.

² A DHCP Relay is needed only when the DHCP Server is on a different subnet from the client.

How to Configure the Configuration Engine

Enabling the CNS Event Agent



Note You must enable the CNS event agent on the device before you enable the CNS configuration agent.

Follow these steps to enable the CNS event agent on the device.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **cns event** {hostname | ip-address} [port-number] [**keepalive** seconds retry-count] [**failover-time** seconds] [**reconnect-time** time] | **backup**]
4. **end**
5. **show running-config**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	cns event {hostname ip-address} [port-number] [keepalive seconds retry-count] [failover-time seconds] [reconnect-time time] backup Example: Device(config)# cns event 10.180.1.27 keepalive 120 10	Enables the event agent, and enters the gateway parameters. <ul style="list-style-type: none"> • For {hostname ip-address}, enter either the hostname or the IP address of the event gateway. • (Optional) For <i>port number</i>, enter the port number for the event gateway. The default port number is 11011. • (Optional) For keepalive <i>seconds</i>, enter how often the device sends keepalive messages. For <i>retry-count</i>, enter the number of unanswered keepalive messages that the device sends before the connection is terminated. The default for each is 0. • (Optional) For failover-time <i>seconds</i>, enter how long the device waits for the primary gateway route after the route to the backup gateway is established. • (Optional) For reconnect-time <i>time</i>, enter the maximum time interval that the device waits before trying to reconnect to the event gateway. • (Optional) Enter backup to show that this is the backup gateway. (If omitted, this is the primary gateway.) <p>Note Though visible in the command-line help string, the encrypt and the clock-timeout <i>time</i> keywords are not supported.</p>
Step 4	end Example: Device(config)# end	Returns to privileged EXEC mode.

	Command or Action	Purpose
Step 5	show running-config Example: Device# <code>show running-config</code>	Verifies your entries.
Step 6	copy running-config startup-config Example: Device# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

What to do next

To verify information about the event agent, use the **show cns event connections** command in privileged EXEC mode.

To disable the CNS event agent, use the **no cns event** { *ip-address* | *hostname* } global configuration command.

Enabling the Cisco IOS CNS Agent

Follow these steps to enable the Cisco IOS CNS agent on the device.

Before you begin

You must enable the CNS event agent on the device before you enable this agent.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **cns config initial** {*hostname* | *ip-address*} [*port-number*]
4. **cns config partial** {*hostname* | *ip-address*} [*port-number*]
5. **end**
6. **show running-config**
7. **copy running-config startup-config**
8. Start the Cisco IOS CNS agent on the device.

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> <code>enable</code>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.

	Command or Action	Purpose
Step 2	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode.
Step 3	cns config initial <i>{hostname ip-address}</i> [<i>port-number</i>] Example: Device(config)# <code>cns config initial 10.180.1.27 10</code>	Enables the Cisco IOS CNS agent, and enters the configuration server parameters. <ul style="list-style-type: none"> • For <i>{hostname ip-address}</i>, enter either the hostname or the IP address of the configuration server. • (Optional) For <i>port number</i>, enter the port number for the configuration server. This command enables the Cisco IOS CNS agent and initiates an initial configuration on the device.
Step 4	cns config partial <i>{hostname ip-address}</i> [<i>port-number</i>] Example: Device(config)# <code>cns config partial 10.180.1.27 10</code>	Enables the Cisco IOS CNS agent, and enters the configuration server parameters. <ul style="list-style-type: none"> • For <i>{hostname ip-address}</i>, enter either the hostname or the IP address of the configuration server. • (Optional) For <i>port number</i>, enter the port number for the configuration server. Enables the Cisco IOS CNS agent and initiates a partial configuration on the device.
Step 5	end Example: Device(config)# <code>end</code>	Returns to privileged EXEC mode.
Step 6	show running-config Example: Device# <code>show running-config</code>	Verifies your entries.
Step 7	copy running-config startup-config Example: Device# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.
Step 8	Start the Cisco IOS CNS agent on the device.	

What to do next

You can now use the Cisco Configuration Engine to remotely send incremental configurations to the device.

Enabling an Initial Configuration for Cisco IOS CNS Agent

Follow these steps to enable the CNS configuration agent and initiate an initial configuration on the device.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **cns template connect** *name*
4. **cli** *config-text*
5. Repeat Steps 3 to 4 to configure another CNS connect template.
6. **exit**
7. **cns connect** *name* [**retries** *number*] [**retry-interval** *seconds*] [**sleep** *seconds*] [**timeout** *seconds*]
8. **discover** {**controller** *controller-type* | **dcli** [**subinterface** *subinterface-number*] | **interface** [*interface-type*] | **line** *line-type*}
9. **template** *name* [... *name*]
10. Repeat Steps 8 to 9 to specify more interface parameters and CNS connect templates in the CNS connect profile.
11. **exit**
12. **hostname** *name*
13. **ip route** *network-number*
14. **cns id** *interface num* {**dns-reverse** | **ipaddress** | **mac-address**} [**event**] [**image**]
15. **cns id** {**hardware-serial** | **hostname** | **string** *string* | **udi**} [**event**] [**image**]
16. **cns config initial** {*hostname* | *ip-address*} [*port-number*] [**event**] [**no-persist**] [**page** *page*] [**source** *ip-address*] [**syntax-check**]
17. **end**
18. **show running-config**
19. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.

	Command or Action	Purpose
Step 3	<p>cns template connect <i>name</i></p> <p>Example:</p> <pre>Device(config)# cns template connect template-dhcp</pre>	Enters CNS template connect configuration mode, and specifies the name of the CNS connect template.
Step 4	<p>cli <i>config-text</i></p> <p>Example:</p> <pre>Device(config-tmpl-conn)# cli ip address dhcp</pre>	Enters a command line for the CNS connect template. Repeat this step for each command line in the template.
Step 5	Repeat Steps 3 to 4 to configure another CNS connect template.	
Step 6	<p>exit</p> <p>Example:</p> <pre>Device(config)# exit</pre>	Returns to global configuration mode.
Step 7	<p>cns connect <i>name</i> [retries <i>number</i>] [retry-interval <i>seconds</i>] [sleep <i>seconds</i>] [timeout <i>seconds</i>]</p> <p>Example:</p> <pre>Device(config)# cns connect dhcp</pre>	<p>Enters CNS connect configuration mode, specifies the name of the CNS connect profile, and defines the profile parameters. The device uses the CNS connect profile to connect to the Configuration Engine.</p> <ul style="list-style-type: none"> • Enter the <i>name</i> of the CNS connect profile. • (Optional) For retries <i>number</i>, enter the number of connection retries. The range is 1 to 30. The default is 3. • (Optional) For retry-interval <i>seconds</i>, enter the interval between successive connection attempts to the Configuration Engine. The range is 1 to 40 seconds. The default is 10 seconds. • (Optional) For sleep <i>seconds</i>, enter the amount of time before which the first connection attempt occurs. The range is 0 to 250 seconds. The default is 0. • (Optional) For timeout <i>seconds</i>, enter the amount of time after which the connection attempts end. The range is 10 to 2000 seconds. The default is 120.
Step 8	<p>discover {controller <i>controller-type</i> dcli [subinterface <i>subinterface-number</i>] interface [<i>interface-type</i>] line <i>line-type</i>}</p> <p>Example:</p> <pre>Device(config-cns-conn)# discover interface gigabitethernet</pre>	<p>Specifies the interface parameters in the CNS connect profile.</p> <ul style="list-style-type: none"> • For controller <i>controller-type</i>, enter the controller type. • For dcli, enter the active data-link connection identifiers (DLCIs).

	Command or Action	Purpose
		<p>(Optional) For subinterface <i>subinterface-number</i>, specify the point-to-point subinterface number that is used to search for active DLCIs.</p> <ul style="list-style-type: none"> • For interface [<i>interface-type</i>], enter the type of interface. • For line <i>line-type</i>, enter the line type.
Step 9	<p>template <i>name</i> [... <i>name</i>]</p> <p>Example:</p> <pre>Device(config-cns-conn)# template template-dhcp</pre>	Specifies the list of CNS connect templates in the CNS connect profile to be applied to the device configuration. You can specify more than one template.
Step 10	Repeat Steps 8 to 9 to specify more interface parameters and CNS connect templates in the CNS connect profile.	
Step 11	<p>exit</p> <p>Example:</p> <pre>Device(config-cns-conn)# exit</pre>	Returns to global configuration mode.
Step 12	<p>hostname <i>name</i></p> <p>Example:</p> <pre>Device(config)# hostname device1</pre>	Enters the hostname for the device.
Step 13	<p>ip route <i>network-number</i></p> <p>Example:</p> <pre>RemoteDevice(config)# ip route 172.28.129.22 255.255.255.255 11.11.11.1</pre>	(Optional) Establishes a static route to the Configuration Engine whose IP address is <i>network-number</i> .
Step 14	<p>cns id <i>interface num</i> {dns-reverse ipaddress mac-address} [event] [image]</p> <p>Example:</p> <pre>RemoteDevice(config)# cns id GigabitEthernet0/1 ipaddress</pre>	<p>(Optional) Sets the unique EventID or ConfigID used by the Configuration Engine. If you enter this command, do not enter the cns id {hardware-serial hostname string string udi} [event] [image] command.</p> <ul style="list-style-type: none"> • For <i>interface num</i>, enter the type of interface. For example, ethernet, group-async, loopback, or virtual-template. This setting specifies from which interface the IP or MAC address should be retrieved to define the unique ID. • For {dns-reverse ipaddress mac-address}, enter dns-reverse to retrieve the hostname and assign it as the unique ID, enter ipaddress to use the IP address, or enter mac-address to use the MAC address as the unique ID.

	Command or Action	Purpose
		<ul style="list-style-type: none"> • (Optional) Enter event to set the ID to be the event-id value used to identify the device. • (Optional) Enter image to set the ID to be the image-id value used to identify the device. <p>Note If both the event and image keywords are omitted, the image-id value is used to identify the device.</p>
Step 15	<p>cns id {hardware-serial hostname string string udi} [event] [image]</p> <p>Example:</p> <pre>RemoteDevice(config)# cns id hostname</pre>	<p>(Optional) Sets the unique EventID or ConfigID used by the Configuration Engine. If you enter this command, do not enter the cns id interface num {dns-reverse ipaddress mac-address} [event] [image] command.</p> <ul style="list-style-type: none"> • For { hardware-serial hostname string string udi }, enter hardware-serial to set the device serial number as the unique ID, enter hostname (the default) to select the device hostname as the unique ID, enter an arbitrary text string for string string as the unique ID, or enter udi to set the unique device identifier (UDI) as the unique ID.
Step 16	<p>cns config initial {<i>hostname</i> <i>ip-address</i>} [<i>port-number</i>] [event] [no-persist] [page page] [source ip-address] [syntax-check]</p> <p>Example:</p> <pre>RemoteDevice(config)# cns config initial 10.1.1.1 no-persist</pre>	<p>Enables the Cisco IOS agent, and initiates an initial configuration.</p> <ul style="list-style-type: none"> • For {<i>hostname</i> <i>ip-address</i>}, enter the hostname or the IP address of the configuration server. • (Optional) For <i>port-number</i>, enter the port number of the configuration server. The default port number is 80. • (Optional) Enable event for configuration success, failure, or warning messages when the configuration is finished. • (Optional) Enable no-persist to suppress the automatic writing to NVRAM of the configuration pulled as a result of entering the cns config initial global configuration command. If the no-persist keyword is not entered, using the cns config initial command causes the resultant configuration to be automatically written to NVRAM. • (Optional) For page page, enter the web page of the initial configuration. The default is /Config/config/asp. • (Optional) Enter source ip-address to use for source IP address.

	Command or Action	Purpose
		<ul style="list-style-type: none"> (Optional) Enable syntax-check to check the syntax when this parameter is entered. <p>Note Though visible in the command-line help string, the encrypt, status url, and inventory keywords are not supported.</p>
Step 17	end Example: <pre>Device(config)# end</pre>	Returns to privileged EXEC mode.
Step 18	show running-config Example: <pre>Device# show running-config</pre>	Verifies your entries.
Step 19	copy running-config startup-config Example: <pre>Device# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

What to do next

To verify information about the configuration agent, use the **show cns config connections** command in privileged EXEC mode.

To disable the CNS Cisco IOS agent, use the **no cns config initial** { *ip-address* | *hostname* } global configuration command.

Refreshing DeviceIDs

Follow these steps to refresh a DeviceID when changing the hostname on the device.

SUMMARY STEPS

- enable**
- show cns config connections**
- Make sure that the CNS event agent is properly connected to the event gateway.
- show cns event connections**
- Record from the output of Step 4 the information for the currently connected connection listed below. You will be using the IP address and port number in subsequent steps of these instructions.
- configure terminal**
- no cns event** *ip-address* *port-number*
- cns event** *ip-address* *port-number*

9. **end**
10. Make sure that you have reestablished the connection between the device and the event connection by examining the output from **show cns event connections**.
11. **show running-config**
12. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	show cns config connections Example: Device# show cns config connections	Displays whether the CNS event agent is connecting to the gateway, connected, or active, and the gateway used by the event agent, its IP address and port number.
Step 3	Make sure that the CNS event agent is properly connected to the event gateway.	Examine the output of show cns config connections for the following: <ul style="list-style-type: none"> • Connection is active. • Connection is using the currently configured device hostname. The DeviceID will be refreshed to correspond to the new hostname configuration using these instructions.
Step 4	show cns event connections Example: Device# show cns event connections	Displays the event connection information for your device.
Step 5	Record from the output of Step 4 the information for the currently connected connection listed below. You will be using the IP address and port number in subsequent steps of these instructions.	
Step 6	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 7	no cns event ip-address port-number Example: Device(config)# no cns event 172.28.129.22 2012	Specifies the IP address and port number that you recorded in Step 5 in this command. This command breaks the connection between the device and the event gateway. It is necessary to first break, then reestablish, this connection to refresh the DeviceID.

	Command or Action	Purpose
Step 8	cns event <i>ip-address port-number</i> Example: Device(config)# cns event 172.28.129.22 2012	Specifies the IP address and port number that you recorded in Step 5 in this command. This command reestablishes the connection between the device and the event gateway.
Step 9	end Example: Device(config)# end	Returns to privileged EXEC mode.
Step 10	Make sure that you have reestablished the connection between the device and the event connection by examining the output from show cns event connections .	
Step 11	show running-config Example: Device# show running-config	Verifies your entries.
Step 12	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Enabling a Partial Configuration for Cisco IOS CNS Agent

Follow these steps to enable the Cisco IOS CNS agent and to initiate a partial configuration on the device.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **cns config partial** {*ip-address | hostname*} [*port-number*] [**source** *ip-address*]
4. **end**
5. **show running-config**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example:	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.

	Command or Action	Purpose
	Device> enable	
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	cns config partial { <i>ip-address</i> <i>hostname</i> } [<i>port-number</i>] [source <i>ip-address</i>] Example: Device(config)# cns config partial 172.28.129.22 2013	Enables the configuration agent, and initiates a partial configuration. <ul style="list-style-type: none"> • For {<i>ip-address</i> <i>hostname</i>}, enter the IP address or the hostname of the configuration server. • (Optional) For <i>port-number</i>, enter the port number of the configuration server. The default port number is 80. • (Optional) Enter source <i>ip-address</i> to use for the source IP address. Note Though visible in the command-line help string, the encrypt keyword is not supported.
Step 4	end Example: Device(config)# end	Returns to privileged EXEC mode.
Step 5	show running-config Example: Device# show running-config	Verifies your entries.
Step 6	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

What to do next

To verify information about the configuration agent, use either the **show cns config stats** or the **show cns config outstanding** command in privileged EXEC mode.

To disable the Cisco IOS agent, use the **no cns config partial** { *ip-address* | *hostname* } global configuration command. To cancel a partial configuration, use the **cns config cancel** global configuration command.

Monitoring CNS Configurations

Table 62: CNS show Commands

Command	Purpose
show cns config connections Device# <code>show cns config connections</code>	Displays the status of the CNS Cisco IOS CNS agent connections.
show cns config outstanding Device# <code>show cns config outstanding</code>	Displays information about incremental (partial) CNS configurations that have started but are not yet completed.
show cns config stats Device# <code>show cns config stats</code>	Displays statistics about the Cisco IOS CNS agent.
show cns event connections Device# <code>show cns event connections</code>	Displays the status of the CNS event agent connections.
show cns event gateway Device# <code>show cns event gateway</code>	Displays the event gateway information for your device.
show cns event stats Device# <code>show cns event stats</code>	Displays statistics about the CNS event agent.
show cns event subject Device# <code>show cns event subject</code>	Displays a list of event agent subjects that are subscribed to by applications.

Additional References

Related Documents

Related Topic	Document Title
Configuration Engine Setup	<i>Cisco Configuration Engine Installation and Setup Guide, 1.5 for Linux</i> https://www.cisco.com/en/US/docs/net_mgmt/configuration_engine/1.5/installation_linux/guide/setup_1.html

Error Message Decoder

Description	Link
To help you research and resolve system error messages in this release, use the Error Message Decoder tool.	https://www.cisco.com/cgi-bin/Support/Errordecoder/index.cgi

Standards and RFCs

Standard/RFC	Title
None	-

MIBs

MIB	MIBs Link
All the supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
<p>The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies.</p> <p>To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds.</p> <p>Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.</p>	http://www.cisco.com/support

Feature History and Information for the Configuration Engine

Release	Modification
Cisco IOS Release 15.0(2)EX1	This feature was introduced.



CHAPTER 35

Configuring the Cisco Discovery Protocol

Cisco Discovery Protocol is a Layer 2, media-independent, and network-independent protocol that runs on Cisco devices and enables networking applications to learn about directly connected devices nearby. This protocol facilitates the management of Cisco devices by discovering these devices, determining how they are configured, and allowing systems using different network-layer protocols to learn about each other.

This module describes Cisco Discovery Protocol Version 2 and how it functions with SNMP.

- [Information About CDP, on page 605](#)
- [How to Configure CDP, on page 606](#)
- [Monitoring and Maintaining Cisco Discovery Protocol, on page 613](#)
- [Additional References, on page 614](#)
- [Feature History and Information for Cisco Discovery Protocol, on page 615](#)

Information About CDP

Cisco Discovery Protocol Overview

Cisco Discovery Protocol is a device discovery protocol that runs over Layer 2 (the data-link layer) on all Cisco-manufactured devices (routers, bridges, access servers, controllers, and switches) and allows network management applications to discover Cisco devices that are neighbors of already known devices. With Cisco Discovery Protocol, network management applications can learn the device type and the SNMP agent address of neighboring devices running lower-layer, transparent protocols. This feature enables applications to send SNMP queries to neighboring devices.

Cisco Discovery Protocol runs on all media that support Subnetwork Access Protocol (SNAP). Because Cisco Discovery Protocol runs over the data-link layer only, two systems that support different network-layer protocols can learn about each other.

Each Cisco Discovery Protocol-configured device sends periodic messages to a multicast address, advertising at least one address at which it can receive SNMP messages. The advertisements also contain time-to-live, or holdtime information, which is the length of time a receiving device holds Cisco Discovery Protocol information before discarding it. Each device also listens to the messages sent by other devices to learn about neighboring devices.

On the device, Cisco Discovery Protocol enables Network Assistant to display a graphical view of the network. The device uses Cisco Discovery Protocol to find cluster candidates and maintain information about cluster members and other devices up to three cluster-enabled devices away from the command device by default.

- Cisco Discovery Protocol identifies connected endpoints that communicate directly with the device.
- To prevent duplicate reports of neighboring devices, only one wired device reports the location information.
- The wired device and the endpoints both send and receive location information.

CDP and Stacks

A device stack appears as a single device in the network. Therefore, CDP discovers the device stack, not the individual stack members. The device stack sends CDP messages to neighboring network devices when there are changes to the device stack membership, such as stack members being added or removed.

Default Cisco Discovery Protocol Configuration

This table shows the default Cisco Discovery Protocol configuration.

Feature	Default Setting
Cisco Discovery Protocol global state	Enabled
Cisco Discovery Protocol interface state	Enabled
Cisco Discovery Protocol timer (packet update frequency)	60 seconds
Cisco Discovery Protocol holdtime (before discarding)	180 seconds
Cisco Discovery Protocol Version-2 advertisements	Enabled

How to Configure CDP

Configuring Cisco Discovery Protocol Characteristics

You can configure these Cisco Discovery Protocol characteristics:

- Frequency of Cisco Discovery Protocol updates
- Amount of time to hold the information before discarding it
- Whether or not to send Version 2 advertisements



Note Steps 3 through 5 are all optional and can be performed in any order.

Follow these steps to configure the Cisco Discovery Protocol characteristics.

SUMMARY STEPS

1. **enable**
2. **configure terminal**

3. `cdp timer seconds`
4. `cdp holdtime seconds`
5. `cdp advertise-v2`
6. `end`
7. `show running-config`
8. `copy running-config startup-config`

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> <code>enable</code>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode.
Step 3	cdp timer seconds Example: Device(config)# <code>cdp timer 20</code>	(Optional) Sets the transmission frequency of Cisco Discovery Protocol updates in seconds. The range is 5 to 254; the default is 60 seconds.
Step 4	cdp holdtime seconds Example: Device(config)# <code>cdp holdtime 60</code>	(Optional) Specifies the amount of time a receiving device should hold the information sent by your device before discarding it. The range is 10 to 255 seconds; the default is 180 seconds.
Step 5	cdp advertise-v2 Example: Device(config)# <code>cdp advertise-v2</code>	(Optional) Configures Cisco Discovery Protocol to send Version 2 advertisements. This is the default state.
Step 6	end Example: Device(config)# <code>end</code>	Returns to privileged EXEC mode.
Step 7	show running-config Example: Device# <code>show running-config</code>	Verifies your entries.

	Command or Action	Purpose
Step 8	copy running-config startup-config Example: Device# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

What to do next

Use the **no** form of the Cisco Discovery Protocol commands to return to the default settings.

Disabling Cisco Discovery Protocol

Cisco Discovery Protocol is enabled by default.



Note Device clusters and other Cisco devices (such as Cisco IP Phones) regularly exchange Cisco Discovery Protocol messages. Disabling Cisco Discovery Protocol can interrupt cluster discovery and device connectivity.

Follow these steps to disable the Cisco Discovery Protocol device discovery capability.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **no cdp run**
4. **end**
5. **show running-config**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> <code>enable</code>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode.
Step 3	no cdp run Example:	Disables Cisco Discovery Protocol.

	Command or Action	Purpose
	Device(config)# no cdp run	
Step 4	end Example: Device(config)# end	Returns to privileged EXEC mode.
Step 5	show running-config Example: Device# show running-config	Verifies your entries.
Step 6	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

What to do next

You must reenble Cisco Discovery Protocol to use it.

Enabling Cisco Discovery Protocol

Cisco Discovery Protocol is enabled by default.



Note Device clusters and other Cisco devices (such as Cisco IP Phones) regularly exchange Cisco Discovery Protocol messages. Disabling Cisco Discovery Protocol can interrupt cluster discovery and device connectivity.

Follow these steps to enable Cisco Discovery Protocol when it has been disabled.

Before you begin

Cisco Discovery Protocol must be disabled, or it cannot be enabled.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **cdp run**
4. **end**
5. **show running-config**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none">• Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	cdp run Example: Device(config)# cdp run	Enables Cisco Discovery Protocol if it has been disabled.
Step 4	end Example: Device(config)# end	Returns to privileged EXEC mode.
Step 5	show running-config Example: Device# show running-config	Verifies your entries.
Step 6	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

What to do next

Use the **show run all** command to show that Cisco Discovery Protocol has been enabled. If you enter only **show run**, the enabling of Cisco Discovery Protocol may not be displayed.

Disabling Cisco Discovery Protocol on an Interface

Cisco Discovery Protocol is enabled by default on all supported interfaces to send and to receive Cisco Discovery Protocol information.



Note Device clusters and other Cisco devices (such as Cisco IP Phones) regularly exchange Cisco Discovery Protocol messages. Disabling Cisco Discovery Protocol can interrupt cluster discovery and device connectivity.



Note Cisco Discovery Protocol bypass is not supported and may cause a port go into err-disabled state.

Follow these steps to disable Cisco Discovery Protocol on a port.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **no cdp enable**
5. **end**
6. **show running-config**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	interface <i>interface-id</i> Example: Device(config)# interface gigabitethernet 1/0/1	Specifies the interface on which you are disabling Cisco Discovery Protocol, and enters interface configuration mode.
Step 4	no cdp enable Example: Device(config-if)# no cdp enable	Disables Cisco Discovery Protocol on the interface specified in Step 3.
Step 5	end Example:	Returns to privileged EXEC mode.

	Command or Action	Purpose
	Device(config)# end	
Step 6	show running-config Example: Device# show running-config	Verifies your entries.
Step 7	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Enabling Cisco Discovery Protocol on an Interface

Cisco Discovery Protocol is enabled by default on all supported interfaces to send and to receive Cisco Discovery Protocol information.



Note Device clusters and other Cisco devices (such as Cisco IP Phones) regularly exchange Cisco Discovery Protocol messages. Disabling Cisco Discovery Protocol can interrupt cluster discovery and device connectivity.



Note Cisco Discovery Protocol bypass is not supported and may cause a port go into err-disabled state.

Follow these steps to enable Cisco Discovery Protocol on a port on which it has been disabled.

Before you begin

Cisco Discovery Protocol must be disabled on the port that you are trying to Cisco Discovery Protocol enable on, or it cannot be enabled.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **cdp enable**
5. **end**
6. **show running-config**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	interface <i>interface-id</i> Example: Device(config)# interface gigabitethernet1/0/1	Specifies the interface on which you are enabling Cisco Discovery Protocol, and enters interface configuration mode.
Step 4	cdp enable Example: Device(config-if)# cdp enable	Enables Cisco Discovery Protocol on a disabled interface.
Step 5	end Example: Device(config)# end	Returns to privileged EXEC mode.
Step 6	show running-config Example: Device# show running-config	Verifies your entries.
Step 7	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Monitoring and Maintaining Cisco Discovery Protocol

Table 63: Commands for Displaying Cisco Discovery Protocol Information

Command	Description
clear cdp counters	Resets the traffic counters to zero.

Command	Description
clear cdp table	Deletes the Cisco Discovery Protocol table of information about neighbors.
show cdp	Displays global information, such as frequency of transmissions and the holdtime for packets being sent.
show cdp entry <i>entry-name</i> [version] [protocol]	Displays information about a specific neighbor. You can enter an asterisk (*) to display all Cisco Discovery Protocol neighbors, or you can enter the name of the neighbor about which you want information. You can also limit the display to information about the protocols enabled on the specified neighbor or information about the version of software running on the device.
show cdp interface [<i>interface-id</i>]	Displays information about interfaces where Cisco Discovery Protocol is enabled. You can limit the display to the interface about which you want information.
show cdp neighbors [<i>interface-id</i>] [<i>detail</i>]	Displays information about neighbors, including device type, interface type and number, holdtime settings, capabilities, platform, and port ID. You can limit the display to neighbors of a specific interface or expand the display to provide more detailed information.
show cdp traffic	Displays Cisco Discovery Protocol counters, including the number of packets sent and received and checksum errors.

Additional References

Related Documents

Related Topic	Document Title
System Management Commands	<i>Network Management Command Reference, Cisco IOS Release 15.2(2)E</i>

Error Message Decoder

Description	Link
To help you research and resolve system error messages in this release, use the Error Message Decoder tool.	https://www.cisco.com/cgi-bin/Support/Errordecoder/index.cgi

Standards and RFCs

Standard/RFC	Title
None	-

MIBs

MIB	MIBs Link
All supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
<p>The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies.</p> <p>To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds.</p> <p>Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.</p>	http://www.cisco.com/support

Feature History and Information for Cisco Discovery Protocol

Release	Modification
Cisco IOS Release 15.0(2)EX1	This feature was introduced.



CHAPTER 36

Configuring Simple Network Management Protocol

- [Prerequisites for SNMP, on page 617](#)
- [Restrictions for SNMP, on page 619](#)
- [Information About SNMP, on page 619](#)
- [How to Configure SNMP, on page 624](#)
- [Monitoring SNMP Status, on page 638](#)
- [SNMP Examples, on page 638](#)
- [Additional References, on page 639](#)
- [Feature History and Information for Simple Network Management Protocol, on page 640](#)

Prerequisites for SNMP

Supported SNMP Versions

This software release supports the following SNMP versions:

- **SNMPv1**—The Simple Network Management Protocol, a Full Internet Standard, defined in RFC 1157.
- **SNMPv2C** replaces the Party-based Administrative and Security Framework of SNMPv2Classic with the community-string-based Administrative Framework of SNMPv2C while retaining the bulk retrieval and improved error handling of SNMPv2Classic. It has these features:
 - **SNMPv2**—Version 2 of the Simple Network Management Protocol, a Draft Internet Standard, defined in RFCs 1902 through 1907.
 - **SNMPv2C**—The community-string-based Administrative Framework for SNMPv2, an Experimental Internet Protocol defined in RFC 1901.
- **SNMPv3**—Version 3 of the SNMP is an interoperable standards-based protocol defined in RFCs 2273 to 2275. SNMPv3 provides secure access to devices by authenticating and encrypting packets over the network and includes these security features:
 - **Message integrity**—Ensures that a packet was not tampered with in transit.
 - **Authentication**—Determines that the message is from a valid source.

- Encryption—Mixes the contents of a package to prevent it from being read by an unauthorized source.



Note To select encryption, enter the **priv** keyword.

Both SNMPv1 and SNMPv2C use a community-based form of security. The community of managers able to access the agent's MIB is defined by an IP address access control list and password.

SNMPv2C includes a bulk retrieval function and more detailed error message reporting to management stations. The bulk retrieval function retrieves tables and large quantities of information, minimizing the number of round-trips required. The SNMPv2C improved error-handling includes expanded error codes that distinguish different kinds of error conditions; these conditions are reported through a single error code in SNMPv1. Error return codes in SNMPv2C report the error type.

SNMPv3 provides for both security models and security levels. A security model is an authentication strategy set up for a user and the group within which the user resides. A security level is the permitted level of security within a security model. A combination of the security level and the security model determine which security method is used when handling an SNMP packet. Available security models are SNMPv1, SNMPv2C, and SNMPv3.

The following table identifies characteristics and compares different combinations of security models and levels:

Table 64: SNMP Security Models and Levels

Model	Level	Authentication	Encryption	Result
SNMPv1	noAuthNoPriv	Community string	No	Uses a community string match for authentication.
SNMPv2C	noAuthNoPriv	Community string	No	Uses a community string match for authentication.
SNMPv3	noAuthNoPriv	Username	No	Uses a username match for authentication.
SNMPv3	authNoPriv	Message Digest 5 (MD5) or Secure Hash Algorithm (SHA)	No	Provides authentication based on the HMAC-MD5 or HMAC-SHA algorithms.

Model	Level	Authentication	Encryption	Result
SNMPv3	authPriv	MD5 or SHA	Data Encryption Standard (DES) or Advanced Encryption Standard (AES)	<p>Provides authentication based on the HMAC-MD5 or HMAC-SHA algorithms.</p> <p>Allows specifying the User-based Security Model (USM) with these encryption algorithms:</p> <ul style="list-style-type: none"> • DES 56-bit encryption in addition to authentication based on the CBC-DES (DES-56) standard. • 3DES 168-bit encryption • AES 128-bit, 192-bit, or 256-bit encryption

You must configure the SNMP agent to use the SNMP version supported by the management station. Because an agent can communicate with multiple managers, you can configure the software to support communications using SNMPv1, SNMPv2C, or SNMPv3.

Restrictions for SNMP

Version Restrictions

- SNMPv1 does not support informs.

Information About SNMP

SNMP Overview

SNMP is an application-layer protocol that provides a message format for communication between managers and agents. The SNMP system consists of an SNMP manager, an SNMP agent, and a management information

base (MIB). The SNMP manager can be part of a network management system (NMS) such as Cisco Prime Infrastructure. The agent and MIB reside on the device. To configure SNMP on the device, you define the relationship between the manager and the agent.

The SNMP agent contains MIB variables whose values the SNMP manager can request or change. A manager can get a value from an agent or store a value into the agent. The agent gathers data from the MIB, the repository for information about device parameters and network data. The agent can also respond to a manager's requests to get or set data.

An agent can send unsolicited traps to the manager. Traps are messages alerting the SNMP manager to a condition on the network. Traps can mean improper user authentication, restarts, link status (up or down), MAC address tracking, closing of a TCP connection, loss of connection to a neighbor, or other significant events.

SNMP Manager Functions

The SNMP manager uses information in the MIB to perform the operations described in the following table:

Table 65: SNMP Operations

Operation	Description
get-request	Retrieves a value from a specific variable.
get-next-request	Retrieves a value from a variable within a table. ³
get-bulk-request ⁴	Retrieves large blocks of data, such as multiple rows in a table, that would otherwise require the transmission of many small blocks of data.
get-response	Replies to a get-request, get-next-request, and set-request sent by an NMS.
set-request	Stores a value in a specific variable.
trap	An unsolicited message sent by an SNMP agent to an SNMP manager when some event has occurred.

³ With this operation, an SNMP manager does not need to know the exact variable name. A sequential search is performed to find the needed variable from within a table.

⁴ The get-bulk command only works with SNMPv2 or later.

SNMP Agent Functions

The SNMP agent responds to SNMP manager requests as follows:

- Get a MIB variable—The SNMP agent begins this function in response to a request from the NMS. The agent retrieves the value of the requested MIB variable and responds to the NMS with that value.
- Set a MIB variable—The SNMP agent begins this function in response to a message from the NMS. The SNMP agent changes the value of the MIB variable to the value requested by the NMS.

The SNMP agent also sends unsolicited trap messages to notify an NMS that a significant event has occurred on the agent. Examples of trap conditions include, but are not limited to, when a port or module goes up or down, when spanning-tree topology changes occur, and when authentication failures occur.

SNMP Community Strings

SNMP community strings authenticate access to MIB objects and function as embedded passwords. In order for the NMS to access the device, the community string definitions on the NMS must match at least one of the three community string definitions on the device.

A community string can have one of the following attributes:

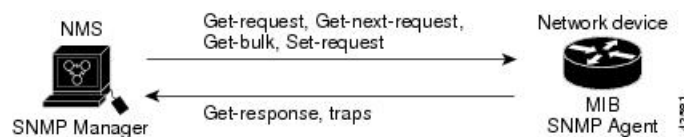
- Read-only (RO)—Gives all objects in the MIB except the community strings read access to authorized management stations, but does not allow write access.
- Read-write (RW)—Gives all objects in the MIB read and write access to authorized management stations, but does not allow access to the community strings.
- When a cluster is created, the command device manages the exchange of messages among member devices and the SNMP application. The Network Assistant software appends the member device number (@esN, where N is the device number) to the first configured RW and RO community strings on the command device and propagates them to the member devices.

SNMP MIB Variables Access

An example of an NMS is the Cisco Prime Infrastructure network management software. Cisco Prime Infrastructure software uses the device MIB variables to set device variables and to poll devices on the network for specific information. The results of a poll can be displayed as a graph and analyzed to troubleshoot internetworking problems, increase network performance, verify the configuration of devices, monitor traffic loads, and more.

As shown in the figure, the SNMP agent gathers data from the MIB. The agent can send traps, or notification of certain events, to the SNMP manager, which receives and processes the traps. Traps alert the SNMP manager to a condition on the network such as improper user authentication, restarts, link status (up or down), MAC address tracking, and so forth. The SNMP agent also responds to MIB-related queries sent by the SNMP manager in *get-request*, *get-next-request*, and *set-request* format.

Figure 62: SNMP Network



SNMP Notifications

SNMP allows the device to send notifications to SNMP managers when particular events occur. SNMP notifications can be sent as traps or inform requests. In command syntax, unless there is an option in the command to select either traps or informs, the keyword traps refers to either traps or informs, or both. Use the **snmp-server host** command to specify whether to send SNMP notifications as traps or informs.



Note SNMPv1 does not support informs.

Traps are unreliable because the receiver does not send an acknowledgment when it receives a trap, and the sender cannot determine if the trap was received. When an SNMP manager receives an inform request, it acknowledges the message with an SNMP response protocol data unit (PDU). If the sender does not receive a response, the inform request can be sent again. Because they can be resent, informs are more likely than traps to reach their intended destination.

The characteristics that make informs more reliable than traps also consume more resources in the device and in the network. Unlike a trap, which is discarded as soon as it is sent, an inform request is held in memory until a response is received or the request times out. Traps are sent only once, but an inform might be resent or retried several times. The retries increase traffic and contribute to a higher overhead on the network. Therefore, traps and informs require a trade-off between reliability and resources. If it is important that the SNMP manager receive every notification, use inform requests. If traffic on the network or memory in the device is a concern and notification is not required, use traps.

SNMP ifIndex MIB Object Values

In an NMS, the IF-MIB generates and assigns an interface index (ifIndex) object value that is a unique number greater than zero to identify a physical or a logical interface. When the device reboots or the device software is upgraded, the device uses this same value for the interface. For example, if the device assigns a port 2 an ifIndex value of 10003, this value is the same after the device reboots.

The device uses one of the values in the following table to assign an ifIndex value to an interface:

Table 66: ifIndex Values

Interface Type	ifIndex Range
SVI ⁵	1–4999
EtherChannel	5001–5048
Tunnel	5078–5142
Physical (such as Gigabit Ethernet or SFP ⁶ -module interfaces) based on type and port numbers	10000–14500
Null	14501
Loopback and Tunnel	24567+

⁵ SVI = switch virtual interface

⁶ SFP = small form-factor pluggable

Default SNMP Configuration

Feature	Default Setting
SNMP agent	Disabled ⁷ .
SNMP trap receiver	None configured.
SNMP traps	None enabled except the trap for TCP connections (tty).
SNMP version	If no version keyword is present, the default is Version 1.

Feature	Default Setting
SNMPv3 authentication	If no keyword is entered, the default is the noauth (noAuthNoPriv) security level.
SNMP notification type	If no type is specified, all notifications are sent.

⁷ This is the default when the device starts and the startup configuration does not have any **snmp-server** global configuration commands.

SNMP Configuration Guidelines

If the device starts and the device startup configuration has at least one **snmp-server** global configuration command, the SNMP agent is enabled.

An SNMP *group* is a table that maps SNMP users to SNMP views. An SNMP *user* is a member of an SNMP group. An SNMP *host* is the recipient of an SNMP trap operation. An SNMP *engine ID* is a name for the local or remote SNMP engine.

When configuring SNMP, follow these guidelines:

- When configuring an SNMP group, do not specify a notify view. The **snmp-server host** global configuration command auto-generates a notify view for the user and then adds it to the group associated with that user. Modifying the group's notify view affects all users associated with that group.
- To configure a remote user, specify the IP address or port number for the remote SNMP agent of the device where the user resides.
- Before you configure remote users for a particular agent, configure the SNMP engine ID, using the **snmp-server engineID** global configuration command with the **remote** option. The remote agent's SNMP engine ID and user password are used to compute the authentication and privacy digests. If you do not configure the remote engine ID first, the configuration command fails.
- When configuring SNMP informs, you need to configure the SNMP engine ID for the remote agent in the SNMP database before you can send proxy requests or informs to it.
- If a local user is not associated with a remote host, the device does not send informs for the **auth** (authNoPriv) and the **priv** (authPriv) authentication levels.
- Changing the value of the SNMP engine ID has significant results. A user's password (entered on the command line) is converted to an MD5 or SHA security digest based on the password and the local engine ID. The command-line password is then destroyed, as required by RFC 2274. Because of this deletion, if the value of the engine ID changes, the security digests of SNMPv3 users become invalid, and you need to reconfigure SNMP users by using the **snmp-server user username** global configuration command. Similar restrictions require the reconfiguration of community strings when the engine ID changes.

How to Configure SNMP

Disabling the SNMP Agent

The **no snmp-server** global configuration command disables all running versions (Version 1, Version 2C, and Version 3) of the SNMP agent on the device. You reenable all versions of the SNMP agent by the first **snmp-server** global configuration command that you enter. There is no Cisco IOS command specifically designated for enabling SNMP.

Follow these steps to disable the SNMP agent.

Before you begin

The SNMP Agent must be enabled before it can be disabled. The SNMP agent is enabled by the first **snmp-server** global configuration command entered on the device.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **no snmp-server**
4. **end**
5. **show running-config**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	no snmp-server Example: Device(config)# no snmp-server	Disables the SNMP agent operation.
Step 4	end Example:	Returns to privileged EXEC mode.

	Command or Action	Purpose
	Device(config)# end	
Step 5	show running-config Example: Device# show running-config	Verifies your entries.
Step 6	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Configuring Community Strings

You use the SNMP community string to define the relationship between the SNMP manager and the agent. The community string acts like a password to permit access to the agent on the device. Optionally, you can specify one or more of these characteristics associated with the string:

- An access list of IP addresses of the SNMP managers that are permitted to use the community string to gain access to the agent
- A MIB view, which defines the subset of all MIB objects accessible to the given community
- Read and write or read-only permission for the MIB objects accessible to the community

Follow these steps to configure a community string on the device.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **snmp-server community** *string* [**view** *view-name*] [**ro** | **rw**] [*access-list-number*]
4. **access-list** *access-list-number* {**deny** | **permit**} *source* [*source-wildcard*]
5. **end**
6. **show running-config**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.

	Command or Action	Purpose
Step 2	<p>configure terminal</p> <p>Example:</p> <pre>Device# configure terminal</pre>	Enters global configuration mode.
Step 3	<p>snmp-server community <i>string</i> [view <i>view-name</i>] [ro rw] [<i>access-list-number</i>]</p> <p>Example:</p> <pre>Device(config)# snmp-server community comaccess ro 4</pre>	<p>Configures the community string.</p> <p>Note The @ symbol is used for delimiting the context information. Avoid using the @ symbol as part of the SNMP community string when configuring this command.</p> <ul style="list-style-type: none"> • For <i>string</i>, specify a string that acts like a password and permits access to the SNMP protocol. You can configure one or more community strings of any length. • (Optional) For view, specify the view record accessible to the community. • (Optional) Specify either read-only (ro) if you want authorized management stations to retrieve MIB objects, or specify read-write (rw) if you want authorized management stations to retrieve and modify MIB objects. By default, the community string permits read-only access to all objects. • (Optional) For <i>access-list-number</i>, enter an IP standard access list numbered from 1 to 99 and 1300 to 1999.
Step 4	<p>access-list <i>access-list-number</i> {deny permit} <i>source</i> [<i>source-wildcard</i>]</p> <p>Example:</p> <pre>Device(config)# access-list 4 deny any</pre>	<p>(Optional) If you specified an IP standard access list number in Step 3, then create the list, repeating the command as many times as necessary.</p> <ul style="list-style-type: none"> • For <i>access-list-number</i>, enter the access list number specified in Step 3. • The deny keyword denies access if the conditions are matched. The permit keyword permits access if the conditions are matched. • For <i>source</i>, enter the IP address of the SNMP managers that are permitted to use the community string to gain access to the agent. • (Optional) For <i>source-wildcard</i>, enter the wildcard bits in dotted decimal notation to be applied to the source. Place ones in the bit positions that you want to ignore. <p>Recall that the access list is always terminated by an implicit deny statement for everything.</p>

	Command or Action	Purpose
Step 5	end Example: Device(config)# end	Returns to privileged EXEC mode.
Step 6	show running-config Example: Device# show running-config	Verifies your entries.
Step 7	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

What to do next

To disable access for an SNMP community, set the community string for that community to the null string (do not enter a value for the community string).

To remove a specific community string, use the **no snmp-server** community string global configuration command.

You can specify an identification name (engine ID) for the local or remote SNMP server engine on the device. You can configure an SNMP server group that maps SNMP users to SNMP views, and you can add new users to the SNMP group.

Configuring SNMP Groups and Users

You can specify an identification name (engine ID) for the local or remote SNMP server engine on the device. You can configure an SNMP server group that maps SNMP users to SNMP views, and you can add new users to the SNMP group.

Follow these steps to configure SNMP groups and users on the device.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **snmp-server engineID** {local *engineid-string* | remote *ip-address* [**udp-port** *port-number*] *engineid-string*}
4. **snmp-server group** *group-name* {**v1** | **v2c** | **v3** {**auth** | **noauth** | **priv**}} [**read** *readview*] [**write** *writeview*] [**notify** *notifyview*] [**access** *access-list*]
5. **snmp-server user** *username* *group-name* {remote *host* [**udp-port** *port*]} {**v1** [**access** *access-list*] | **v2c** [**access** *access-list*] | **v3** [**encrypted**] [**access** *access-list*] [**auth** {**md5** | **sha**} *auth-password*]} [**priv** {**des** | **3des** | **aes** {**128** | **192** | **256**}} *priv-password*]

6. `end`
7. `show running-config`
8. `copy running-config startup-config`

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p><code>enable</code></p> <p>Example:</p> <pre>Device> enable</pre>	<p>Enables privileged EXEC mode.</p> <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	<p><code>configure terminal</code></p> <p>Example:</p> <pre>Device# configure terminal</pre>	<p>Enters global configuration mode.</p>
Step 3	<p><code>snmp-server engineID {local <i>engineid-string</i> remote <i>ip-address</i> [<i>udp-port port-number</i>] <i>engineid-string</i>}</code></p> <p>Example:</p> <pre>Device(config)# snmp-server engineID local 1234</pre>	<p>Configures a name for either the local or remote copy of SNMP.</p> <ul style="list-style-type: none"> • The <i>engineid-string</i> is a 24-character ID string with the name of the copy of SNMP. You need not specify the entire 24-character engine ID if it has trailing zeros. Specify only the portion of the engine ID up to the point where only zeros remain in the value. The Step Example configures an engine ID of 123400000000000000000000. • If you select remote, specify the <i>ip-address</i> of the device that contains the remote copy of SNMP and the optional User Datagram Protocol (UDP) port on the remote device. The default is 162.
Step 4	<p><code>snmp-server group <i>group-name</i> {v1 v2c v3 {auth noauth priv}} [<i>read readview</i>] [<i>write writeview</i>] [<i>notify notifyview</i>] [<i>access access-list</i>]</code></p> <p>Example:</p> <pre>Device(config)# snmp-server group public v2c access lmnop</pre>	<p>Configures a new SNMP group on the remote device.</p> <p>For <i>group-name</i>, specify the name of the group.</p> <p>Specify one of the following security models:</p> <ul style="list-style-type: none"> • v1 is the least secure of the possible security models. • v2c is the second least secure model. It allows transmission of informs and integers twice the normal width. • v3, the most secure, requires you to select one of the following authentication levels: <ul style="list-style-type: none"> auth—Enables the Message Digest 5 (MD5) and the Secure Hash Algorithm (SHA) packet authentication.

	Command or Action	Purpose
		<p>noauth—Enables the noAuthNoPriv security level. This is the default if no keyword is specified.</p> <p>priv—Enables Data Encryption Standard (DES) packet encryption (also called privacy).</p> <p>(Optional) Enter read <i>readview</i> with a string (not to exceed 64 characters) that is the name of the view in which you can only view the contents of the agent.</p> <p>(Optional) Enter write <i>writeview</i> with a string (not to exceed 64 characters) that is the name of the view in which you enter data and configure the contents of the agent.</p> <p>(Optional) Enter notify <i>notifyview</i> with a string (not to exceed 64 characters) that is the name of the view in which you specify a notify, inform, or trap.</p> <p>(Optional) Enter access <i>access-list</i> with a string (not to exceed 64 characters) that is the name of the access list.</p>
<p>Step 5</p>	<p>snmp-server user <i>username group-name</i> { remote <i>host</i> [udp-port <i>port</i>] } { v1 [access <i>access-list</i>] v2c [access <i>access-list</i>] v3 [encrypted] [access <i>access-list</i>] [auth { md5 sha } <i>auth-password</i>] } [priv { des 3des aes { 128 192 256 } } <i>priv-password</i>]</p> <p>Example:</p> <pre>Device(config)# snmp-server user Pat public v2c</pre>	<p>Adds a new user for an SNMP group.</p> <p>The <i>username</i> is the name of the user on the host that connects to the agent.</p> <p>The <i>group-name</i> is the name of the group to which the user is associated.</p> <p>Enter remote to specify a remote SNMP entity to which the user belongs and the hostname or IP address of that entity with the optional UDP port number. The default is 162.</p> <p>Enter the SNMP version number (v1, v2c, or v3). If you enter v3, you have these additional options:</p> <ul style="list-style-type: none"> • encrypted specifies that the password appears in encrypted format. This keyword is available only when the v3 keyword is specified. • auth is an authentication level setting session that can be either the HMAC-MD5-96 (md5) or the HMAC-SHA-96 (sha) authentication level and requires a password string <i>auth-password</i> (not to exceed 64 characters). <p>If you enter v3 you can also configure a private (priv) encryption algorithm and password string <i>priv-password</i> using the following keywords (not to exceed 64 characters):</p> <ul style="list-style-type: none"> • priv specifies the User-based Security Model (USM). • des specifies the use of the 56-bit DES algorithm. • 3des specifies the use of the 168-bit DES algorithm.

	Command or Action	Purpose
		<ul style="list-style-type: none"> aes specifies the use of the DES algorithm. You must select either 128-bit, 192-bit, or 256-bit encryption. (Optional) Enter access <i>access-list</i> with a string (not to exceed 64 characters) that is the name of the access list.
Step 6	end Example: Device(config) # end	Returns to privileged EXEC mode.
Step 7	show running-config Example: Device# show running-config	Verifies your entries.
Step 8	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Configuring SNMP Notifications

A trap manager is a management station that receives and processes traps. Traps are system alerts that the device generates when certain events occur. By default, no trap manager is defined, and no traps are sent. Devices running this Cisco IOS release can have an unlimited number of trap managers.



Note Many commands use the word **traps** in the command syntax. Unless there is an option in the command to select either traps or informs, the keyword **traps** refers to traps, informs, or both. Use the **snmp-server host** global configuration command to specify whether to send SNMP notifications as traps or informs.

You can use the **snmp-server host** global configuration command for a specific host to receive the notification types listed in the following table. You can enable any or all of these traps and configure a trap manager to receive them.

Table 67: Device Notification Types

Notification Type Keyword	Description
bridge	Generates STP bridge MIB traps.
cluster	Generates a trap when the cluster configuration changes.
config	Generates a trap for SNMP configuration changes.
copy-config	Generates a trap for SNMP copy configuration changes.

Notification Type Keyword	Description
cpu threshold	Allow CPU-related traps.
entity	Generates a trap for SNMP entity changes.
envmon	Generates environmental monitor traps. You can enable any or all of these environmental traps: fan, shutdown, status, supply, temperature.
errdisable	Generates a trap for a port VLAN errdisabled. You can also set a maximum trap rate per minute. The range is from 0 to 10000; the default is 0, which means there is no rate limit.
flash	Generates SNMP FLASH notifications. In a device stack, you can optionally enable notification for flash insertion or removal, which would cause a trap to be issued whenever a device in the stack is removed or inserted (physical removal, power cycle, or reload).
fru-ctrl	Generates entity field-replaceable unit (FRU) control traps. In the device stack, this trap refers to the insertion or removal of a device in the stack.
hsrp	Generates a trap for Hot Standby Router Protocol (HSRP) changes.
ipmulticast	Generates a trap for IP multicast routing changes.
ipsla	Generates a trap for the SNMP IP Service Level Agreements (SLAs).
mac-notification	Generates a trap for MAC address notifications.
msdp	Generates a trap for Multicast Source Discovery Protocol (MSDP) changes.
ospf	Generates a trap for Open Shortest Path First (OSPF) changes. You can enable any or all of these traps: Cisco specific, errors, link-state advertisement, rate limit, retransmit, and state changes.
pim	Generates a trap for Protocol-Independent Multicast (PIM) changes. You can enable any or all of these traps: invalid PIM messages, neighbor changes, and rendezvous point (RP)-mapping changes.
port-security	<p>Generates SNMP port security traps. You can also set a maximum trap rate per second. The range is from 0 to 1000; the default is 0, which means that there is no rate limit.</p> <p>Note When you configure a trap by using the notification type port-security, configure the port security trap first, and then configure the port security trap rate:</p> <ol style="list-style-type: none"> snmp-server enable traps port-security snmp-server enable traps port-security trap-rate rate
snmp	Generates a trap for SNMP-type notifications for authentication, cold start, warm start, link up or link down.

Notification Type Keyword	Description
storm-control	Generates a trap for SNMP storm-control. You can also set a maximum trap rate per minute. The range is from 0 to 1000; the default is 0 (no limit is imposed; a trap is sent at every occurrence).
stpx	Generates SNMP STP Extended MIB traps.
syslog	Generates SNMP syslog traps.
tty	Generates a trap for TCP connections. This trap is enabled by default.
vlan-membership	Generates a trap for SNMP VLAN membership changes.
vlancreate	Generates SNMP VLAN created traps.
vlandelete	Generates SNMP VLAN deleted traps.
vtp	Generates a trap for VLAN Trunking Protocol (VTP) changes.

Follow these steps to configure the device to send traps or informs to a host.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **snmp-server engineID remote ip-address engineid-string**
4. **snmp-server user username group-name {remote host [udp-port port]} {v1 [access access-list] | v2c [access access-list] | v3 [encrypted] [access access-list] [auth {md5 | sha} auth-password]}**
5. **snmp-server group group-name {v1 | v2c | v3 {auth | noauth | priv}} [read readview] [write writeview] [notify notifyview] [access access-list]**
6. **snmp-server host host-addr [informs | traps] [version {1 | 2c | 3 {auth | noauth | priv}}] community-string [notification-type]**
7. **snmp-server enable traps notification-types**
8. **snmp-server trap-source interface-id**
9. **snmp-server queue-length length**
10. **snmp-server trap-timeout seconds**
11. **end**
12. **show running-config**
13. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.

	Command or Action	Purpose
Step 2	<p>configure terminal</p> <p>Example:</p> <pre>Device# configure terminal</pre>	Enters global configuration mode.
Step 3	<p>snmp-server engineID remote <i>ip-address engineid-string</i></p> <p>Example:</p> <pre>Device(config)# snmp-server engineID remote 192.180.1.27 00000063000100a1c0b4011b</pre>	Specifies the engine ID for the remote host.
Step 4	<p>snmp-server user <i>username group-name</i> { remote host [udp-port <i>port</i>] } { v1 [access <i>access-list</i>] v2c [access <i>access-list</i>] v3 [encrypted] [access <i>access-list</i>] [auth { md5 sha } <i>auth-password</i>] }</p> <p>Example:</p> <pre>Device(config)# snmp-server user Pat public v2c</pre>	<p>Configures an SNMP user to be associated with the remote host created in Step 3.</p> <p>Note You cannot configure a remote user for an address without first configuring the engine ID for the remote host. Otherwise, you receive an error message, and the command is not executed.</p>
Step 5	<p>snmp-server group <i>group-name</i> { v1 v2c v3 { auth noauth priv } } [read <i>readview</i>] [write <i>writeview</i>] [notify <i>notifyview</i>] [access <i>access-list</i>]</p> <p>Example:</p> <pre>Device(config)# snmp-server group public v2c access lmnop</pre>	Configures an SNMP group.
Step 6	<p>snmp-server host <i>host-addr</i> [informs traps] [version { 1 2c 3 { auth noauth priv } }] <i>community-string</i> [<i>notification-type</i>]</p> <p>Example:</p> <pre>Device(config)# snmp-server host 203.0.113.1 comaccess snmp</pre>	<p>Specifies the recipient of an SNMP trap operation.</p> <p>For <i>host-addr</i>, specify the name or Internet address of the host (the targeted recipient).</p> <p>(Optional) Specify traps (the default) to send SNMP traps to the host.</p> <p>(Optional) Specify informs to send SNMP informs to the host.</p> <p>(Optional) Specify the SNMP version (1, 2c, or 3). SNMPv1 does not support informs.</p> <p>(Optional) For Version 3, select authentication level auth, noauth, or priv.</p> <p>Note The priv keyword is available only when the cryptographic software image is installed.</p> <p>For <i>community-string</i>, when version 1 or version 2c is specified, enter the password-like community string sent with the notification operation. When version 3 is specified, enter the SNMPv3 username.</p>

	Command or Action	Purpose
		<p>The @ symbol is used for delimiting the context information. Avoid using the @ symbol as part of the SNMP community string when configuring this command.</p> <p>(Optional) For <i>notification-type</i>, use the keywords listed in the table above. If no type is specified, all notifications are sent.</p>
Step 7	<p>snmp-server enable traps <i>notification-types</i></p> <p>Example:</p> <pre>Device(config)# snmp-server enable traps snmp</pre>	<p>Enables the device to send traps or informs and specifies the type of notifications to be sent. For a list of notification types, see the table above, or enter snmp-server enable traps ?</p> <p>To enable multiple types of traps, you must enter a separate snmp-server enable traps command for each trap type.</p> <p>Note When you configure a trap by using the notification type port-security, configure the port security trap first, and then configure the port security trap rate:</p> <ol style="list-style-type: none"> a. snmp-server enable traps port-security b. snmp-server enable traps port-security trap-rate <i>rate</i>
Step 8	<p>snmp-server trap-source <i>interface-id</i></p> <p>Example:</p> <pre>Device(config)# snmp-server trap-source gigabitethernet 1/0/1</pre>	<p>(Optional) Specifies the source interface, which provides the IP address for the trap message. This command also sets the source IP address for informs.</p>
Step 9	<p>snmp-server queue-length <i>length</i></p> <p>Example:</p> <pre>Device(config)# snmp-server queue-length 20</pre>	<p>(Optional) Establishes the message queue length for each trap host. The range is 1 to 5000; the default is 10.</p>
Step 10	<p>snmp-server trap-timeout <i>seconds</i></p> <p>Example:</p> <pre>Device(config)# snmp-server trap-timeout 60</pre>	<p>(Optional) Defines how often to resend trap messages. The range is 1 to 1000; the default is 30 seconds.</p>
Step 11	<p>end</p> <p>Example:</p> <pre>Device(config)# end</pre>	<p>Returns to privileged EXEC mode.</p>
Step 12	<p>show running-config</p> <p>Example:</p> <pre>Device# show running-config</pre>	<p>Verifies your entries.</p>

	Command or Action	Purpose
Step 13	copy running-config startup-config Example: Device# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

What to do next

The **snmp-server host** command specifies which hosts receive the notifications. The **snmp-server enable traps** command globally enables the method for the specified notification (for traps and informs). To enable a host to receive an inform, you must configure an **snmp-server host informs** command for the host and globally enable informs by using the **snmp-server enable traps** command.

To remove the specified host from receiving traps, use the **no snmp-server host** *host* global configuration command. The **no snmp-server host** command with no keywords disables traps, but not informs, to the host. To disable informs, use the **no snmp-server host informs** global configuration command. To disable a specific trap type, use the **no snmp-server enable traps** *notification-types* global configuration command.

Setting the Agent Contact and Location Information

Follow these steps to set the system contact and location of the SNMP agent so that these descriptions can be accessed through the configuration file.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **snmp-server contact** *text*
4. **snmp-server location** *text*
5. **end**
6. **show running-config**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> <code>enable</code>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode.

	Command or Action	Purpose
Step 3	snmp-server contact <i>text</i> Example: Device(config)# snmp-server contact Dial System Operator at beeper 21555	Sets the system contact string.
Step 4	snmp-server location <i>text</i> Example: Device(config)# snmp-server location Building 3/Room 222	Sets the system location string.
Step 5	end Example: Device(config)# end	Returns to privileged EXEC mode.
Step 6	show running-config Example: Device# show running-config	Verifies your entries.
Step 7	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Limiting TFTP Servers Used Through SNMP

Follow these steps to limit the TFTP servers used for saving and loading configuration files through SNMP to the servers specified in an access list.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **snmp-server tftp-server-list** *access-list-number*
4. **access-list** *access-list-number* {**deny** | **permit**} *source* [*source-wildcard*]
5. **end**
6. **show running-config**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	snmp-server tftp-server-list access-list-number Example: Device(config)# snmp-server tftp-server-list 44	Limits the TFTP servers used for configuration file copies through SNMP to the servers in the access list. For <i>access-list-number</i> , enter an IP standard access list numbered from 1 to 99 and 1300 to 1999.
Step 4	access-list access-list-number {deny permit} source [source-wildcard] Example: Device(config)# access-list 44 permit 10.1.1.2	Creates a standard access list, repeating the command as many times as necessary. For <i>access-list-number</i> , enter the access list number specified in Step 3. The deny keyword denies access if the conditions are matched. The permit keyword permits access if the conditions are matched. For <i>source</i> , enter the IP address of the TFTP servers that can access the device. (Optional) For <i>source-wildcard</i> , enter the wildcard bits, in dotted decimal notation, to be applied to the source. Place ones in the bit positions that you want to ignore. The access list is always terminated by an implicit deny statement for everything.
Step 5	end Example: Device(config)# end	Returns to privileged EXEC mode.
Step 6	show running-config Example: Device# show running-config	Verifies your entries.

	Command or Action	Purpose
Step 7	copy running-config startup-config Example: Device# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Monitoring SNMP Status

To display SNMP input and output statistics, including the number of illegal community string entries, errors, and requested variables, use the **show snmp** privileged EXEC command. You also can use the other privileged EXEC commands listed in the table to display SNMP information.

Table 68: Commands for Displaying SNMP Information

Command	Purpose
show snmp	Displays SNMP statistics.
	Displays information on the local SNMP engine and all remote engines that have been configured on the device.
show snmp group	Displays information on each SNMP group on the network.
show snmp pending	Displays information on pending SNMP requests.
show snmp sessions	Displays information on the current SNMP sessions.
show snmp user	Displays information on each SNMP user name in the SNMP users table. Note You must use this command to display SNMPv3 configuration information for auth noauth priv mode. This information is not displayed in the show running-config output.

SNMP Examples

This example shows how to enable all versions of SNMP. The configuration permits any SNMP manager to access all objects with read-only permissions using the community string *public*. This configuration does not cause the device to send any traps.

```
Device(config)# snmp-server community public
```

This example shows how to permit any SNMP manager to access all objects with read-only permission using the community string *public*. The device also sends VTP traps to the hosts 192.180.1.111 and 192.180.1.33

using SNMPv1 and to the host 192.180.1.27 using SNMPv2C. The community string *public* is sent with the traps.

```
Device(config)# snmp-server community public
Device(config)# snmp-server enable traps vtp
Device(config)# snmp-server host 192.180.1.27 version 2c public
Device(config)# snmp-server host 192.180.1.111 version 1 public
Device(config)# snmp-server host 192.180.1.33 public
```

This example shows how to allow read-only access for all objects to members of access list 4 that use the *comaccess* community string. No other SNMP managers have access to any objects. SNMP Authentication Failure traps are sent by SNMPv2C to the host *cisco.com* using the community string *public*.

```
Device(config)# snmp-server community comaccess ro 4
Device(config)# snmp-server enable traps snmp authentication
Device(config)# snmp-server host cisco.com version 2c public
```

This example shows how to send Entity MIB traps to the host *cisco.com*. The community string is restricted. The first line enables the device to send Entity MIB traps in addition to any traps previously enabled. The second line specifies the destination of these traps and overwrites any previous **snmp-server** host commands for the host *cisco.com*.

```
Device(config)# snmp-server enable traps entity
Device(config)# snmp-server host cisco.com restricted entity
```

This example shows how to enable the device to send all traps to the host *myhost.cisco.com* using the community string *public*:

```
Device(config)# snmp-server enable traps
Device(config)# snmp-server host myhost.cisco.com public
```

This example shows how to associate a user with a remote host and to send **auth** (authNoPriv) authentication-level informs when the user enters global configuration mode:

```
Device(config)# snmp-server engineID remote 192.180.1.27 00000063000100a1c0b4011b
Device(config)# snmp-server group authgroup v3 auth
Device(config)# snmp-server user authuser authgroup remote 192.180.1.27 v3 auth md5 mypassword
Device(config)# snmp-server user authuser authgroup v3 auth md5 mypassword
Device(config)# snmp-server host 192.180.1.27 informs version 3 auth authuser config
Device(config)# snmp-server enable traps
Device(config)# snmp-server inform retries 0
```

Additional References

Related Documents

Related Topic	Document Title
SNMP Commands	<i>Network Management Command Reference, Cisco IOS Release 15.2(2)E</i>

Error Message Decoder

Description	Link
To help you research and resolve system error messages in this release, use the Error Message Decoder tool.	https://www.cisco.com/cgi-bin/Support/Errordecoder/index.cgi

Standards and RFCs

Standard/RFC	Title
None	-

MIBs

MIB	MIBs Link
All supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
<p>The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies.</p> <p>To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds.</p> <p>Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.</p>	http://www.cisco.com/support

Feature History and Information for Simple Network Management Protocol

Release	Modification
Cisco IOS Release 15.0(2)EX1	This feature was introduced.



CHAPTER 37

Configuring SPAN and RSPAN

- [Prerequisites for SPAN and RSPAN, on page 641](#)
- [Restrictions for SPAN and RSPAN, on page 641](#)
- [Information About SPAN and RSPAN, on page 643](#)
- [How to Configure SPAN and RSPAN, on page 653](#)
- [Monitoring SPAN and RSPAN Operations, on page 675](#)
- [SPAN and RSPAN Configuration Examples, on page 676](#)
- [Additional References, on page 678](#)
- [Feature History and Information for SPAN and RSPAN, on page 679](#)

Prerequisites for SPAN and RSPAN

SPAN

- You can limit SPAN traffic to specific VLANs by using the **filter vlan** keyword. If a trunk port is being monitored, only traffic on the VLANs specified with this keyword is monitored. By default, all VLANs are monitored on a trunk port.

RSPAN

- We recommend that you configure an RSPAN VLAN before you configure an RSPAN source or a destination session.

Restrictions for SPAN and RSPAN

SPAN

The restrictions for SPAN are as follows:

- On each device, you can configure 66 sessions. A maximum of 4 source sessions can be configured and the remaining sessions can be configured as RSPAN destinations sessions. A source session is either a local SPAN session or an RSPAN source session.
- For SPAN sources, you can monitor traffic for a single port or VLAN or a series or range of ports or VLANs for each session. You cannot mix source ports and source VLANs within a single SPAN session.

- The destination port cannot be a source port; a source port cannot be a destination port.
- You cannot have two SPAN sessions using the same destination port.
- When you configure a device port as a SPAN destination port, it is no longer a normal device port; only monitored traffic passes through the SPAN destination port.
- Entering SPAN configuration commands does not remove previously configured SPAN parameters. You must enter the **no monitor session** {*session_number* | **all** | **local** | **remote**} global configuration command to delete configured SPAN parameters.
- For local SPAN, outgoing packets through the SPAN destination port carry the original encapsulation headers—untagged, ISL, or IEEE 802.1Q—if the **encapsulation replicate** keywords are specified. If the keywords are not specified, the packets are sent in native form.
- You can configure a disabled port to be a source or destination port, but the SPAN function does not start until the destination port and at least one source port or source VLAN are enabled.
- You cannot mix source VLANs and filter VLANs within a single SPAN session.

Traffic monitoring in a SPAN session has the following restrictions:

- Sources can be ports or VLANs, but you cannot mix source ports and source VLANs in the same session.
- Wireshark does not capture egress packets when egress span is active.
- You can run both a local SPAN and an RSPAN source session in the same device or device stack. The device or device stack supports a total of 66 source and RSPAN destination sessions.
- You can configure two separate SPAN or RSPAN source sessions with separate or overlapping sets of SPAN source ports and VLANs. Both switched and routed ports can be configured as SPAN sources and destinations.
- You can have multiple destination ports in a SPAN session, but no more than 64 destination ports per device stack.
- SPAN sessions do not interfere with the normal operation of the device. However, an oversubscribed SPAN destination, for example, a 10-Mb/s port monitoring a 100-Mb/s port, can result in dropped or lost packets.
- When SPAN or RSPAN is enabled, each packet being monitored is sent twice, once as normal traffic and once as a monitored packet. Monitoring a large number of ports or VLANs could potentially generate large amounts of network traffic.
- You can configure SPAN sessions on disabled ports; however, a SPAN session does not become active unless you enable the destination port and at least one source port or VLAN for that session.
- The device does not support a combination of local SPAN and RSPAN in a single session.
 - An RSPAN source session cannot have a local destination port.
 - An RSPAN destination session cannot have a local source port.
 - An RSPAN destination session and an RSPAN source session that are using the same RSPAN VLAN cannot run on the same device or device stack.

RSPAN

The restrictions for RSPAN are as follows:

- RSPAN does not support BPDU packet monitoring or other Layer 2 device protocols.
- The RSPAN VLAN is configured only on trunk ports and not on access ports. To avoid unwanted traffic in RSPAN VLANs, make sure that the VLAN remote-span feature is supported in all the participating devices.
- RSPAN VLANs are included as sources for port-based RSPAN sessions when source trunk ports have active RSPAN VLANs. RSPAN VLANs can also be sources in SPAN sessions. However, since the device does not monitor spanned traffic, it does not support egress spanning of packets on any RSPAN VLAN identified as the destination of an RSPAN source session on the device.
- CDP packets are not forwarded in RSPAN configured VLAN due to limitation in hardware. The workaround is to disable CDP on all the interfaces carrying RSPAN VLAN on the devices connected to the switch.
- If you enable VTP and VTP pruning, RSPAN traffic is pruned in the trunks to prevent the unwanted flooding of RSPAN traffic across the network for VLAN IDs that are lower than 1005.
- To use RSPAN, the switch must be running the LAN Base image.

Information About SPAN and RSPAN

SPAN and RSPAN

You can analyze network traffic passing through ports or VLANs by using SPAN or RSPAN to send a copy of the traffic to another port on the device or on another device that has been connected to a network analyzer or other monitoring or security device. SPAN copies (or mirrors) traffic received or sent (or both) on source ports or source VLANs to a destination port for analysis. SPAN does not affect the switching of network traffic on the source ports or VLANs. You must dedicate the destination port for SPAN use. Except for traffic that is required for the SPAN or RSPAN session, destination ports do not receive or forward traffic.

Only traffic that enters or leaves source ports or traffic that enters or leaves source VLANs can be monitored by using SPAN; traffic routed to a source VLAN cannot be monitored. For example, if incoming traffic is being monitored, traffic that gets routed from another VLAN to the source VLAN cannot be monitored; however, traffic that is received on the source VLAN and routed to another VLAN can be monitored.

You can use the SPAN or RSPAN destination port to inject traffic from a network security device. For example, if you connect a Cisco Intrusion Detection System (IDS) sensor appliance to a destination port, the IDS device can send TCP reset packets to close down the TCP session of a suspected attacker.

Local SPAN

Local SPAN supports a SPAN session entirely within one device; all source ports or source VLANs and destination ports are in the same device or device stack. Local SPAN copies traffic from one or more source ports in any VLAN or from one or more VLANs to a destination port for analysis.

Figure 63: Example of Local SPAN Configuration on a Single Device

All traffic on port 5 (the source port) is mirrored to port 10 (the destination port). A network analyzer on port 10 receives all network traffic from port 5 without being physically attached to port

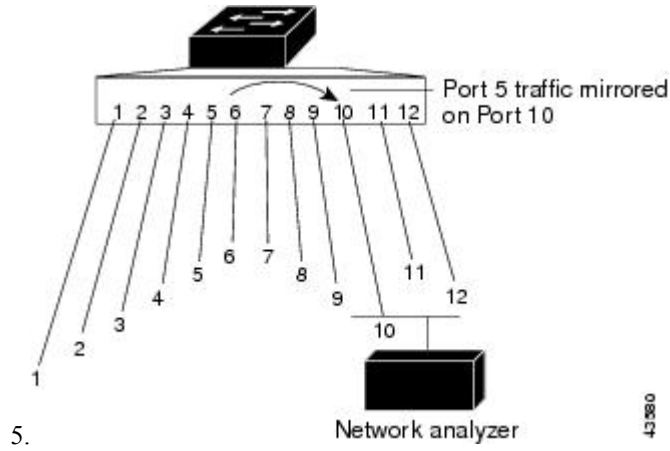
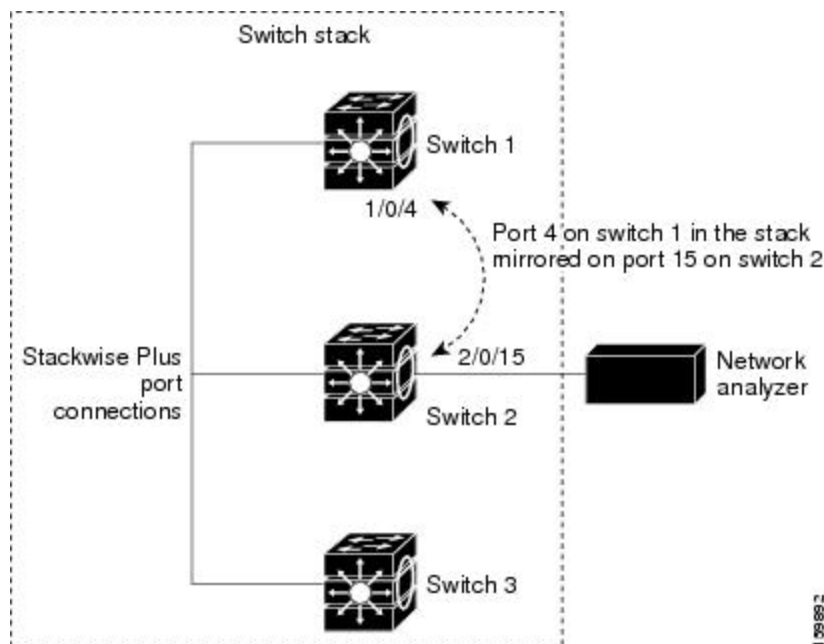


Figure 64: Example of Local SPAN Configuration on a Device Stack

This is an example of a local SPAN in a device stack, where the source and destination ports reside on different stack members.



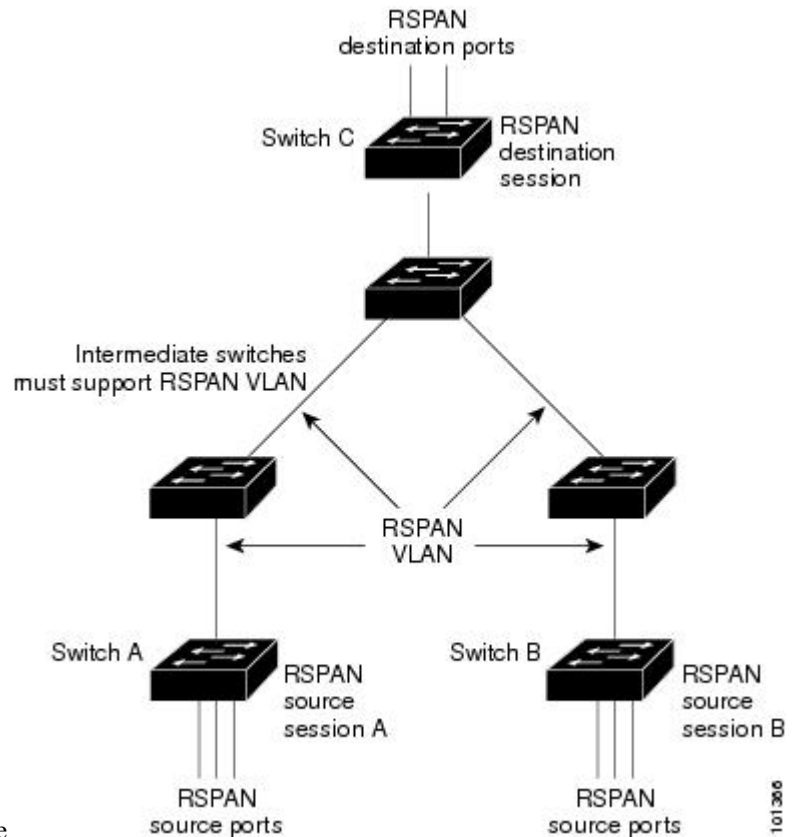
Remote SPAN

RSPAN supports source ports, source VLANs, and destination ports on different devices (or different device stacks), enabling remote monitoring of multiple devices across your network.

Figure 65: Example of RSPAN Configuration

The figure below shows source ports on Device A and Device B. The traffic for each RSPAN session is carried over a user-specified RSPAN VLAN that is dedicated for that RSPAN session in all participating devices.

The RSPAN traffic from the source ports or VLANs is copied into the RSPAN VLAN and forwarded over trunk ports carrying the RSPAN VLAN to a destination session monitoring the RSPAN VLAN. Each RSPAN source device must have either ports or VLANs as RSPAN sources. The destination is always a physical port,



as shown on Device C in the figure.

SPAN and RSPAN Concepts and Terminology

SPAN Sessions

SPAN sessions (local or remote) allow you to monitor traffic on one or more ports, or one or more VLANs, and send the monitored traffic to one or more destination ports.

A local SPAN session is an association of a destination port with source ports or source VLANs, all on a single network device. Local SPAN does not have separate source and destination sessions. Local SPAN sessions gather a set of ingress and egress packets specified by the user and form them into a stream of SPAN data, which is directed to the destination port.

RSPAN consists of at least one RSPAN source session, an RSPAN VLAN, and at least one RSPAN destination session. You separately configure RSPAN source sessions and RSPAN destination sessions on different network devices. To configure an RSPAN source session on a device, you associate a set of source ports or source VLANs with an RSPAN VLAN. The output of this session is the stream of SPAN packets that are sent to the RSPAN VLAN. To configure an RSPAN destination session on another device, you associate the destination port with the RSPAN VLAN. The destination session collects all RSPAN VLAN traffic and sends it out the RSPAN destination port.

An RSPAN source session is very similar to a local SPAN session, except for where the packet stream is directed. In an RSPAN source session, SPAN packets are relabeled with the RSPAN VLAN ID and directed over normal trunk ports to the destination device.

An RSPAN destination session takes all packets received on the RSPAN VLAN, strips off the VLAN tagging, and presents them on the destination port. The session presents a copy of all RSPAN VLAN packets (except Layer 2 control packets) to the user for analysis.

More than one source session and more than one destination session can be active in the same RSPAN VLAN. Intermediate devices also can separate the RSPAN source and destination sessions. These devices are unable to run RSPAN, but they must respond to the requirements of the RSPAN VLAN.

Traffic monitoring in a SPAN session has these restrictions:

- Sources can be ports or VLANs, but you cannot mix source ports and source VLANs in the same session.
- You can run both a local SPAN and an RSPAN source session in the same device or device stack. The device or device stack supports a total of 66 source and RSPAN destination sessions.
- You can configure two separate SPAN or RSPAN source sessions with separate or overlapping sets of SPAN source ports and VLANs. Both switched and routed ports can be configured as SPAN sources and destinations.
- You can have multiple destination ports in a SPAN session, but no more than 64 destination ports per device stack.
- SPAN sessions do not interfere with the normal operation of the device. However, an oversubscribed SPAN destination, for example, a 10-Mb/s port monitoring a 100-Mb/s port, can result in dropped or lost packets.
- When SPAN or RSPAN is enabled, each packet being monitored is sent twice, once as normal traffic and once as a monitored packet. Therefore monitoring a large number of ports or VLANs could potentially generate large amounts of network traffic.
- You can configure SPAN sessions on disabled ports; however, a SPAN session does not become active unless you enable the destination port and at least one source port or VLAN for that session.
- The device does not support a combination of local SPAN and RSPAN in a single session.
 - An RSPAN source session cannot have a local destination port.
 - An RSPAN destination session cannot have a local source port.
 - An RSPAN destination session and an RSPAN source session that are using the same RSPAN VLAN cannot run on the same device or device stack.

Monitored Traffic

SPAN sessions can monitor these traffic types:

- Receive (Rx) SPAN—Receive (or ingress) SPAN monitors as much as possible all of the packets received by the source interface or VLAN before any modification or processing is performed by the device. A copy of each packet received by the source is sent to the destination port for that SPAN session.

Packets that are modified because of routing or Quality of Service (QoS)—for example, modified Differentiated Services Code Point (DSCP)—are copied before modification.

Features that can cause a packet to be dropped during receive processing have no effect on ingress SPAN; the destination port receives a copy of the packet even if the actual incoming packet is dropped. These features include IP standard and extended input Access Control Lists (ACLs), ingress QoS policing, VLAN ACLs, and egress QoS policing.

- **Transmit (Tx) SPAN**—Transmit (or egress) SPAN monitors as much as possible all of the packets sent by the source interface after all modification and processing is performed by the device. A copy of each packet sent by the source is sent to the destination port for that SPAN session. The copy is provided after the packet is modified.

Packets that are modified because of routing (for example, with modified time-to-live (TTL), MAC address, or QoS values) are duplicated (with the modifications) at the destination port.

Features that can cause a packet to be dropped during transmit processing also affect the duplicated copy for SPAN. These features include IP standard and extended output ACLs and egress QoS policing.

- **Both**—In a SPAN session, you can also monitor a port or VLAN for both received and sent packets. This is the default.

The default configuration for local SPAN session ports is to send all packets untagged. SPAN also does not normally monitor bridge protocol data unit (BPDU) packets and Layer 2 protocols, such as Cisco Discovery Protocol (CDP), VLAN Trunk Protocol (VTP), Dynamic Trunking Protocol (DTP), Spanning Tree Protocol (STP), and Port Aggregation Protocol (PAgP). However, when you enter the **encapsulation replicate** keywords when configuring a destination port, these changes occur:

- Packets are sent on the destination port with the same encapsulation (untagged or IEEE 802.1Q) that they had on the source port.
- Packets of all types, including BPDU and Layer 2 protocol packets, are monitored.

Therefore, a local SPAN session with **encapsulation replicate** enabled can have a mixture of untagged and IEEE 802.1Q tagged packets appear on the destination port.

Device congestion can cause packets to be dropped at ingress source ports, egress source ports, or SPAN destination ports. In general, these characteristics are independent of one another. For example:

- A packet might be forwarded normally but dropped from monitoring due to an oversubscribed SPAN destination port.
- An ingress packet might be dropped from normal forwarding, but still appear on the SPAN destination port.
- An egress packet dropped because of device congestion is also dropped from egress SPAN.

In some SPAN configurations, multiple copies of the same source packet are sent to the SPAN destination port. For example, a bidirectional (both Rx and Tx) SPAN session is configured for the Rx monitor on port A and Tx monitor on port B. If a packet enters the device through port A and is switched to port B, both incoming and outgoing packets are sent to the destination port. Both packets are the same unless a Layer 3 rewrite occurs, in which case the packets are different because of the packet modification.

Source Ports

A source port (also called a monitored port) is a switched or routed port that you monitor for network traffic analysis.

In a local SPAN session or RSPAN source session, you can monitor source ports or VLANs for traffic in one or both directions.

The device supports any number of source ports (up to the maximum number of available ports on the device) and any number of source VLANs (up to the maximum number of VLANs supported).

However, the device supports a maximum of (local or RSPAN) with source ports or VLANs. You cannot mix ports and VLANs in a single session.

A source port has these characteristics:

- It can be monitored in multiple SPAN sessions.
- Each source port can be configured with a direction (ingress, egress, or both) to monitor.
- It can be any port type (for example, EtherChannel, Gigabit Ethernet, and so forth).
- For EtherChannel sources, you can monitor traffic for the entire EtherChannel or individually on a physical port as it participates in the port channel.
- It can be an access port, trunk port, routed port, or voice VLAN port.
- It cannot be a destination port.
- Source ports can be in the same or different VLANs.
- You can monitor multiple source ports in a single session.

Source VLANs

VLAN-based SPAN (VSPAN) is the monitoring of the network traffic in one or more VLANs. The SPAN or RSPAN source interface in VSPAN is a VLAN ID, and traffic is monitored on all the ports for that VLAN.

VSPAN has these characteristics:

- All active ports in the source VLAN are included as source ports and can be monitored in either or both directions.
- On a given port, only traffic on the monitored VLAN is sent to the destination port.
- If a destination port belongs to a source VLAN, it is excluded from the source list and is not monitored.
- If ports are added to or removed from the source VLANs, the traffic on the source VLAN received by those ports is added to or removed from the sources being monitored.
- You cannot use filter VLANs in the same session with VLAN sources.
- You can monitor only Ethernet VLANs.

VLAN Filtering

When you monitor a trunk port as a source port, by default, all VLANs active on the trunk are monitored. You can limit SPAN traffic monitoring on trunk source ports to specific VLANs by using VLAN filtering.

- VLAN filtering applies only to trunk ports or to voice VLAN ports.
- VLAN filtering applies only to port-based sessions and is not allowed in sessions with VLAN sources.
- When a VLAN filter list is specified, only those VLANs in the list are monitored on trunk ports or on voice VLAN access ports.
- SPAN traffic coming from other port types is not affected by VLAN filtering; that is, all VLANs are allowed on other ports.
- VLAN filtering affects only traffic forwarded to the destination SPAN port and does not affect the switching of normal traffic.

Destination Port

Each local SPAN session or RSPAN destination session must have a destination port (also called a monitoring port) that receives a copy of traffic from the source ports or VLANs and sends the SPAN packets to the user, usually a network analyzer.

A destination port has these characteristics:

- For a local SPAN session, the destination port must reside on the same device or device stack as the source port. For an RSPAN session, it is located on the device containing the RSPAN destination session. There is no destination port on a device or device stack running only an RSPAN source session.
- When a port is configured as a SPAN destination port, the configuration overwrites the original port configuration. When the SPAN destination configuration is removed, the port reverts to its previous configuration. If a configuration change is made to the port while it is acting as a SPAN destination port, the change does not take effect until the SPAN destination configuration had been removed.



Note When QoS is configured on the SPAN destination port, QoS takes effect immediately.

- If the port was in an EtherChannel group, it is removed from the group while it is a destination port. If it was a routed port, it is no longer a routed port.
- It can be any Ethernet physical port.
- It cannot be a secure port.
- It cannot be a source port.
- It can participate in only one SPAN session at a time (a destination port in one SPAN session cannot be a destination port for a second SPAN session).
- When it is active, incoming traffic is disabled. The port does not transmit any traffic except that required for the SPAN session. Incoming traffic is never learned or forwarded on a destination port.
- If ingress traffic forwarding is enabled for a network security device, the destination port forwards traffic at Layer 2.
- It does not participate in any of the Layer 2 protocols (STP, VTP, CDP, DTP, PagP).
- A destination port that belongs to a source VLAN of any SPAN session is excluded from the source list and is not monitored.
- The maximum number of destination ports in a device or device stack is 64.

Local SPAN and RSPAN destination ports function differently with VLAN tagging and encapsulation:

- For local SPAN, if the **encapsulation replicate** keywords are specified for the destination port, these packets appear with the original encapsulation (untagged, ISL, or IEEE 802.1Q). If these keywords are not specified, packets appear in the untagged format. Therefore, the output of a local SPAN session with **encapsulation replicate** enabled can contain a mixture of untagged, ISL, or IEEE 802.1Q-tagged packets.
- For RSPAN, the original VLAN ID is lost because it is overwritten by the RSPAN VLAN identification. Therefore, all packets appear on the destination port as untagged.

RSPAN VLAN

The RSPAN VLAN carries SPAN traffic between RSPAN source and destination sessions. RSPAN VLAN has these special characteristics:

- All traffic in the RSPAN VLAN is always flooded.
- No MAC address learning occurs on the RSPAN VLAN.
- RSPAN VLAN traffic only flows on trunk ports.
- RSPAN VLANs must be configured in VLAN configuration mode by using the **remote-span** VLAN configuration mode command.
- STP can run on RSPAN VLAN trunks but not on SPAN destination ports.
- An RSPAN VLAN cannot be a private-VLAN primary or secondary VLAN.

For VLANs 1 to 1005 that are visible to VLAN Trunking Protocol (VTP), the VLAN ID and its associated RSPAN characteristic are propagated by VTP. If you assign an RSPAN VLAN ID in the extended VLAN range (1006 to 4094), you must manually configure all intermediate devices.

It is normal to have multiple RSPAN VLANs in a network at the same time with each RSPAN VLAN defining a network-wide RSPAN session. That is, multiple RSPAN source sessions anywhere in the network can contribute packets to the RSPAN session. It is also possible to have multiple RSPAN destination sessions throughout the network, monitoring the same RSPAN VLAN and presenting traffic to the user. The RSPAN VLAN ID separates the sessions.

SPAN and RSPAN Interaction with Other Features

SPAN interacts with these features:

- Routing—SPAN does not monitor routed traffic. VSPAN only monitors traffic that enters or exits the device, not traffic that is routed between VLANs. For example, if a VLAN is being Rx-monitored and the device routes traffic from another VLAN to the monitored VLAN, that traffic is not monitored and not received on the SPAN destination port.
- STP—A destination port does not participate in STP while its SPAN or RSPAN session is active. The destination port can participate in STP after the SPAN or RSPAN session is disabled. On a source port, SPAN does not affect the STP status. STP can be active on trunk ports carrying an RSPAN VLAN.
- CDP—A SPAN destination port does not participate in CDP while the SPAN session is active. After the SPAN session is disabled, the port again participates in CDP.
- VTP—You can use VTP to prune an RSPAN VLAN between devices.
- VLAN and trunking—You can modify VLAN membership or trunk settings for source or destination ports at any time. However, changes in VLAN membership or trunk settings for a destination port do not take effect until you remove the SPAN destination configuration. Changes in VLAN membership or trunk settings for a source port immediately take effect, and the respective SPAN sessions automatically adjust accordingly.
- EtherChannel—You can configure an EtherChannel group as a source port a SPAN destination port. When a group is configured as a SPAN source, the entire group is monitored.

If a physical port is added to a monitored EtherChannel group, the new port is added to the SPAN source port list. If a port is removed from a monitored EtherChannel group, it is automatically removed from the source port list.

A physical port that belongs to an EtherChannel group can be configured as a SPAN source port and still be a part of the EtherChannel. In this case, data from the physical port is monitored as it participates in the EtherChannel. However, if a physical port that belongs to an EtherChannel group is configured as a SPAN destination, it is removed from the group. After the port is removed from the SPAN session, it rejoins the EtherChannel group. Ports removed from an EtherChannel group remain members of the group, but they are in the inactive or suspended state.

If a physical port that belongs to an EtherChannel group is a destination port and the EtherChannel group is a source, the port is removed from the EtherChannel group and from the list of monitored ports.

- Multicast traffic can be monitored. For egress and ingress port monitoring, only a single unedited packet is sent to the SPAN destination port. It does not reflect the number of times the multicast packet is sent.
- A private-VLAN port cannot be a SPAN destination port.
- A secure port cannot be a SPAN destination port.

For SPAN sessions, do not enable port security on ports with monitored egress when ingress forwarding is enabled on the destination port. For RSPAN source sessions, do not enable port security on any ports with monitored egress.

- An IEEE 802.1x port can be a SPAN source port. You can enable IEEE 802.1x on a port that is a SPAN destination port; however, IEEE 802.1x is disabled until the port is removed as a SPAN destination.

For SPAN sessions, do not enable IEEE 802.1x on ports with monitored egress when ingress forwarding is enabled on the destination port. For RSPAN source sessions, do not enable IEEE 802.1x on any ports that are egress monitored.

SPAN and RSPAN and Device Stacks

Because the stack of devices represents one logical device, local SPAN source ports and destination ports can be in different devices in the stack. Therefore, the addition or deletion of devices in the stack can affect a local SPAN session, as well as an RSPAN source or destination session. An active session can become inactive when a device is removed from the stack or an inactive session can become active when a device is added to the stack.

Flow-Based SPAN

You can control the type of network traffic to be monitored in SPAN or RSPAN sessions by using flow-based SPAN (FSPAN) or flow-based RSPAN (FRSPAN), which apply access control lists (ACLs) to the monitored traffic on the source ports. The FSPAN ACLs can be configured to filter IPv4, IPv6, and non-IP monitored traffic.

You apply an ACL to a SPAN session through the interface. It is applied to all the traffic that is monitored on all interfaces in the SPAN session. The packets that are permitted by this ACL are copied to the SPAN destination port. No other packets are copied to the SPAN destination port.

The original traffic continues to be forwarded, and any port, VLAN, and router ACLs attached are applied. The FSPAN ACL does not have any effect on the forwarding decisions. Similarly, the port, VLAN, and router ACLs do not have any effect on the traffic monitoring. If a security input ACL denies a packet and it is not forwarded, the packet is still copied to the SPAN destination ports if the FSPAN ACL permits it. But if the security output ACL denies a packet and it is not sent, it is not copied to the SPAN destination ports. However, if the security output ACL permits the packet to go out, it is only copied to the SPAN destination ports if the FSPAN ACL permits it. This is also true for an RSPAN session.

You can attach three types of FSPAN ACLs to the SPAN session:

- IPv4 FSPAN ACL— Filters only IPv4 packets.
- IPv6 FSPAN ACL— Filters only IPv6 packets.
- MAC FSPAN ACL— Filters only non-IP packets.

The security ACLs have higher priority than the FSPAN ACLs on a device. If FSPAN ACLs are applied, and you later add more security ACLs that cannot fit in the hardware memory, the FSPAN ACLs that you applied are removed from memory to allow space for the security ACLs. A system message notifies you of this action, which is called unloading. When there is again space for the FSPAN ACLs to reside in memory, they are added to the hardware memory on the device. A system message notifies you of this action, which is called reloading. The IPv4, IPv6 and MAC FSPAN ACLs can be unloaded or reloaded independently.

If a VLAN-based FSPAN session configured on a stack cannot fit in the hardware memory on one or more devices, it is treated as unloaded on those devices, and traffic meant for the FSPAN ACL and sourcing on that device is not copied to the SPAN destination ports. The FSPAN ACL continues to be correctly applied, and traffic is copied to the SPAN destination ports on the devices where the FSPAN ACL fits in the hardware memory.

When an empty FSPAN ACL is attached, some hardware functions copy all traffic to the SPAN destination ports for that ACL. If sufficient hardware resources are not available, even an empty FSPAN ACL can be unloaded.

IPv4 and MAC FSPAN ACLs are supported on all feature sets. IPv6 FSPAN ACLs are supported only in the advanced IP Services feature set.

Default SPAN and RSPAN Configuration

Table 69: Default SPAN and RSPAN Configuration

Feature	Default Setting
SPAN state (SPAN and RSPAN)	Disabled.
Source port traffic to monitor	Both received and sent traffic (both).
Encapsulation type (destination port)	Native form (untagged packets).
Ingress forwarding (destination port)	Disabled.
VLAN filtering	On a trunk interface used as a source port, all VLANs are monitored.
RSPAN VLANs	None configured.

Configuration Guidelines

SPAN Configuration Guidelines

- To remove a source or destination port or VLAN from the SPAN session, use the **no monitor session session_number source {interface interface-id | vlan vlan-id}** global configuration command or the **no**

monitor session *session_number* **destination interface** *interface-id* global configuration command. For destination interfaces, the **encapsulation** options are ignored with the **no** form of the command.

- To monitor all VLANs on the trunk port, use the **no monitor session** *session_number* **filter** global configuration command.

RSPAN Configuration Guidelines

- All the SPAN configuration guidelines apply to RSPAN.
- As RSPAN VLANs have special properties, you should reserve a few VLANs across your network for use as RSPAN VLANs; do not assign access ports to these VLANs.
- You can apply an output ACL to RSPAN traffic to selectively filter or monitor specific packets. Specify these ACLs on the RSPAN VLAN in the RSPAN source devices.
- For RSPAN configuration, you can distribute the source ports and the destination ports across multiple devices in your network.
- Access ports (including voice VLAN ports) on the RSPAN VLAN are put in the inactive state.
- You can configure any VLAN as an RSPAN VLAN as long as these conditions are met:
 - The same RSPAN VLAN is used for an RSPAN session in all the devices.
 - All participating devices support RSPAN.

FSPAN and FRSPAN Configuration Guidelines

- When at least one FSPAN ACL is attached, FSPAN is enabled.
- When you attach at least one FSPAN ACL that is not empty to a SPAN session, and you have not attached one or more of the other FSPAN ACLs (for instance, you have attached an IPv4 ACL that is not empty, and have not attached IPv6 and MAC ACLs), FSPAN blocks the traffic that would have been filtered by the unattached ACLs. Therefore, this traffic is not monitored.

How to Configure SPAN and RSPAN

Creating a Local SPAN Session

Follow these steps to create a SPAN session and specify the source (monitored) ports or VLANs and the destination (monitoring) ports.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **no monitor session** {*session_number* | **all** | **local** | **remote**}
4. **monitor session** *session_number* **source** {**interface** *interface-id* | **vlan** *vlan-id*} [, | -] [**both** | **rx** | **tx**]
5. **monitor session** *session_number* **destination** {**interface** *interface-id* [, | -] [**encapsulation** **replicate**]}

6. `end`
7. `show running-config`
8. `copy running-config startup-config`

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Device> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: <pre>Device# configure terminal</pre>	Enters global configuration mode.
Step 3	no monitor session <i>{session_number all local remote}</i> Example: <pre>Device(config)# no monitor session all</pre>	Removes any existing SPAN configuration for the session. <ul style="list-style-type: none"> • For <i>session_number</i>, the range is 1 to 4. • all—Removes all SPAN sessions. • local—Removes all local sessions. • remote—Removes all remote SPAN sessions.
Step 4	monitor session <i>session_number</i> source <i>{interface interface-id vlan vlan-id}</i> [, -] [both rx tx] Example: <pre>Device(config)# monitor session 1 source interface gigabitethernet1/0/1</pre>	Specifies the SPAN session and the source port (monitored port). <ul style="list-style-type: none"> • For <i>session_number</i>, the range is 1 to 4. • For <i>interface-id</i>, specify the source port to monitor. Valid interfaces include physical interfaces and port-channel logical interfaces (port-channel <i>port-channel-number</i>). Valid port-channel numbers are 1 to 6. • For <i>vlan-id</i>, specify the source VLAN to monitor. The range is 1 to 4094 (excluding the RSPAN VLAN). <p>Note A single session can include multiple sources (ports or VLANs) defined in a series of commands, but you cannot combine source ports and source VLANs in one session.</p> • (Optional) [, -] Specifies a series or range of interfaces. Enter a space before and after the comma; enter a space before and after the hyphen.

	Command or Action	Purpose
		<ul style="list-style-type: none"> • (Optional) both rx tx—Specifies the direction of traffic to monitor. If you do not specify a traffic direction, the source interface sends both sent and received traffic. <ul style="list-style-type: none"> • both—Monitors both received and sent traffic. • rx—Monitors received traffic. • tx—Monitors sent traffic. <p>Note You can use the monitor session session_number source command multiple times to configure multiple source ports.</p>
Step 5	<p>monitor session <i>session_number</i> destination {interface <i>interface-id</i> [, -] [encapsulation replicate]}</p> <p>Example:</p> <pre>Device(config)# monitor session 1 destination interface gigabitethernet1/0/2 encapsulation replicate</pre>	<p>Specifies the SPAN session and the destination port (monitoring port). The port LED changes to amber when the configuration changes take effect. The LED returns to its original state(green) only after removing the SPAN destination configuration.</p> <p>Note For local SPAN, you must use the same session number for the source and destination interfaces.</p> <ul style="list-style-type: none"> • For <i>session_number</i>, specify the session number entered in step 4. • For <i>interface-id</i>, specify the destination port. The destination interface must be a physical port; it cannot be an EtherChannel, and it cannot be a VLAN. • (Optional) [, -] Specifies a series or range of interfaces. Enter a space before and after the comma; enter a space before and after the hyphen. <p>(Optional) encapsulation replicate specifies that the destination interface replicates the source interface encapsulation method. If not selected, the default is to send packets in native form (untagged).</p> <p>Note You can use monitor session session_number destination command multiple times to configure multiple destination ports.</p>
Step 6	<p>end</p> <p>Example:</p> <pre>Device(config)# end</pre>	<p>Returns to privileged EXEC mode.</p>
Step 7	<p>show running-config</p>	<p>Verifies your entries.</p>

	Command or Action	Purpose
	Example: Device# <code>show running-config</code>	
Step 8	copy running-config startup-config Example: Device# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Creating a Local SPAN Session and Configuring Incoming Traffic

Follow these steps to create a SPAN session, to specify the source ports or VLANs and the destination ports, and to enable incoming traffic on the destination port for a network security device (such as a Cisco IDS Sensor Appliance).

SUMMARY STEPS

1. `enable`
2. `configure terminal`
3. `no monitor session {session_number | all | local | remote}`
4. `monitor session session_number source {interface interface-id | vlan vlan-id} [, | -] [both | rx | tx]`
5. `monitor session session_number destination {interface interface-id [, | -] [encapsulation replicate[ingress {dot1q vlan vlan-id | untagged vlan vlan-id | vlan vlan-id}]}`
6. `end`
7. `show running-config`
8. `copy running-config startup-config`

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> <code>enable</code>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode.
Step 3	no monitor session {session_number all local remote} Example:	Removes any existing SPAN configuration for the session. <ul style="list-style-type: none"> • For <code>session_number</code>, the range is 1 to 4.

	Command or Action	Purpose
	<pre>Device(config)# no monitor session all</pre>	<ul style="list-style-type: none"> • all—Removes all SPAN sessions. • local—Removes all local sessions. • remote—Removes all remote SPAN sessions.
Step 4	<p>monitor session <i>session_number</i> source {interface <i>interface-id</i> / vlan <i>vlan-id</i>} [, -] [both rx tx]</p> <p>Example:</p> <pre>Device(config)# monitor session 2 source gigabitethernet0/1 rx</pre>	Specifies the SPAN session and the source port (monitored port).
Step 5	<p>monitor session <i>session_number</i> destination {interface <i>interface-id</i> [, -] [encapsulation replicate[ingress {dot1q vlan <i>vlan-id</i> untagged vlan <i>vlan-id</i> vlan <i>vlan-id</i>}]}</p> <p>Example:</p> <pre>Device(config)# monitor session 2 destination interface gigabitethernet1/0/2 encapsulation replicate ingress dot1q vlan 6</pre>	<p>Specifies the SPAN session, the destination port, the packet encapsulation, and the ingress VLAN and encapsulation.</p> <ul style="list-style-type: none"> • For <i>session_number</i>, specify the session number entered in Step 4. • For <i>interface-id</i>, specify the destination port. The destination interface must be a physical port; it cannot be an EtherChannel, and it cannot be a VLAN. • (Optional) [, -]—Specifies a series or range of interfaces. Enter a space before and after the comma or hyphen. • (Optional) encapsulation replicate—Specifies that the destination interface replicates the source interface encapsulation method. If not selected, the default is to send packets in native form (untagged). • ingress—Enables forwarding of incoming traffic on the destination port and to specify the encapsulation type. <ul style="list-style-type: none"> • dot1q vlan <i>vlan-id</i>—Accepts incoming packets with IEEE 802.1Q encapsulation with the specified VLAN as the default VLAN. • untagged vlan <i>vlan-id</i> or vlan <i>vlan-id</i>—Accepts incoming packets with untagged encapsulation type with the specified VLAN as the default VLAN.
Step 6	<p>end</p> <p>Example:</p> <pre>Device(config)# end</pre>	Returns to privileged EXEC mode.

	Command or Action	Purpose
Step 7	show running-config Example: Device# <code>show running-config</code>	Verifies your entries.
Step 8	copy running-config startup-config Example: Device# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Specifying VLANs to Filter

Follow these steps to limit SPAN source traffic to specific VLANs.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **no monitor session** {*session_number* | **all** | **local** | **remote**}
4. **monitor session** *session_number* **source interface** *interface-id*
5. **monitor session** *session_number* **filter vlan** *vlan-id* [, | -]
6. **monitor session** *session_number* **destination** {**interface** *interface-id* [, | -] [**encapsulation replicate**]}
7. **end**
8. **show running-config**
9. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> <code>enable</code>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode.
Step 3	no monitor session { <i>session_number</i> all local remote } Example:	Removes any existing SPAN configuration for the session. <ul style="list-style-type: none"> • For <i>session_number</i>, the range is 1 to 66.

	Command or Action	Purpose
	<pre>Device(config)# no monitor session all</pre>	<ul style="list-style-type: none"> • all—Removes all SPAN sessions. • local—Removes all local sessions. • remote—Removes all remote SPAN sessions.
Step 4	<p>monitor session <i>session_number</i> source interface <i>interface-id</i></p> <p>Example:</p> <pre>Device(config)# monitor session 2 source interface gigabitethernet1/0/2 rx</pre>	<p>Specifies the characteristics of the source port (monitored port) and SPAN session.</p> <ul style="list-style-type: none"> • For <i>session_number</i>, the range is 1 to 66. • For <i>interface-id</i>, specify the source port to monitor. The interface specified must already be configured as a trunk port.
Step 5	<p>monitor session <i>session_number</i> filter vlan <i>vlan-id</i> [, -]</p> <p>Example:</p> <pre>Device(config)# monitor session 2 filter vlan 1 - 5 , 9</pre>	<p>Limits the SPAN source traffic to specific VLANs.</p> <ul style="list-style-type: none"> • For <i>session_number</i>, enter the session number specified in Step 4. • For <i>vlan-id</i>, the range is 1 to 4094. • (Optional) Use a comma (,) to specify a series of VLANs, or use a hyphen (-) to specify a range of VLANs. Enter a space before and after the comma; enter a space before and after the hyphen.
Step 6	<p>monitor session <i>session_number</i> destination {interface <i>interface-id</i> [, -] [encapsulation replicate]}</p> <p>Example:</p> <pre>Device(config)# monitor session 2 destination interface gigabitethernet1/0/1</pre>	<p>Specifies the SPAN session and the destination port (monitoring port).</p> <ul style="list-style-type: none"> • For <i>session_number</i>, specify the session number entered in Step 4. • For <i>interface-id</i>, specify the destination port. The destination interface must be a physical port; it cannot be an EtherChannel, and it cannot be a VLAN. • (Optional) [, -] Specifies a series or range of interfaces. Enter a space before and after the comma; enter a space before and after the hyphen. • (Optional) encapsulation replicate specifies that the destination interface replicates the source interface encapsulation method. If not selected, the default is to send packets in native form (untagged).
Step 7	<p>end</p> <p>Example:</p> <pre>Device(config)# end</pre>	<p>Returns to privileged EXEC mode.</p>

	Command or Action	Purpose
Step 8	show running-config Example: Device# <code>show running-config</code>	Verifies your entries.
Step 9	copy running-config startup-config Example: Device# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Configuring a VLAN as an RSPAN VLAN

Follow these steps to create a new VLAN, then configure it to be the RSPAN VLAN for the RSPAN session.

SUMMARY STEPS

1. `enable`
2. `configure terminal`
3. `vlan vlan-id`
4. `remote-span`
5. `end`
6. `show running-config`
7. `copy running-config startup-config`

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> <code>enable</code>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode.
Step 3	vlan <i>vlan-id</i> Example: Device(config)# <code>vlan 100</code>	Enters a VLAN ID to create a VLAN, or enters the VLAN ID of an existing VLAN, and enters VLAN configuration mode. The range is 2 to 1001 and 1006 to 4094.

	Command or Action	Purpose
		The RSPAN VLAN cannot be VLAN 1 (the default VLAN) or VLAN IDs 1002 through 1005 (reserved for Token Ring and FDDI VLANs).
Step 4	remote-span Example: Device(config-vlan)# remote-span	Configures the VLAN as an RSPAN VLAN.
Step 5	end Example: Device(config-vlan)# end	Returns to privileged EXEC mode.
Step 6	show running-config Example: Device# show running-config	Verifies your entries.
Step 7	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

What to do next

You must create the RSPAN VLAN in all devices that will participate in RSPAN. If the RSPAN VLAN-ID is in the normal range (lower than 1005) and VTP is enabled in the network, you can create the RSPAN VLAN in one device, and VTP propagates it to the other devices in the VTP domain. For extended-range VLANs (greater than 1005), you must configure RSPAN VLAN on both source and destination devices and any intermediate devices.

Use VTP pruning to get an efficient flow of RSPAN traffic, or manually delete the RSPAN VLAN from all trunks that do not need to carry the RSPAN traffic.

To remove the remote SPAN characteristic from a VLAN and convert it back to a normal VLAN, use the **no remote-span** VLAN configuration command.

To remove a source port or VLAN from the SPAN session, use the **no monitor session session_number source {interface interface-id | vlan vlan-id}** global configuration command. To remove the RSPAN VLAN from the session, use the **no monitor session session_number destination remote vlan vlan-id**.

Creating an RSPAN Source Session

Follow these steps to create and start an RSPAN source session and to specify the monitored source and the destination RSPAN VLAN.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **no monitor session** {*session_number* | **all** | **local** | **remote**}
4. **monitor session** *session_number* **source** {**interface** *interface-id* | **vlan** *vlan-id*} [, | -] [**both** | **rx** | **tx**]
5. **monitor session** *session_number* **destination remote vlan** *vlan-id*
6. **end**
7. **show running-config**
8. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	no monitor session { <i>session_number</i> all local remote } Example: Device(config)# no monitor session 1	Removes any existing SPAN configuration for the session. <ul style="list-style-type: none"> • For <i>session_number</i>, the range is 1 to 66. • all—Removes all SPAN sessions. • local—Removes all local sessions. • remote—Removes all remote SPAN sessions.
Step 4	monitor session <i>session_number</i> source { interface <i>interface-id</i> vlan <i>vlan-id</i> } [, -] [both rx tx] Example: Device(config)# monitor session 1 source interface gigabitethernet1/0/1 tx	Specifies the RSPAN session and the source port (monitored port). <ul style="list-style-type: none"> • For <i>session_number</i>, the range is 1 to 66. • Enter a source port or source VLAN for the RSPAN session: <ul style="list-style-type: none"> • For <i>interface-id</i>, specifies the source port to monitor. Valid interfaces include physical interfaces and port-channel logical interfaces (port-channel <i>port-channel-number</i>). Valid port-channel numbers are 1 to 48. • For <i>vlan-id</i>, specifies the source VLAN to monitor. The range is 1 to 4094 (excluding the RSPAN VLAN).

	Command or Action	Purpose
		<p>A single session can include multiple sources (ports or VLANs), defined in a series of commands, but you cannot combine source ports and source VLANs in one session.</p> <ul style="list-style-type: none"> • (Optional) <code>[, -]</code>—Specifies a series or range of interfaces. Enter a space before and after the comma; enter a space before and after the hyphen. • (Optional) both rx tx—Specifies the direction of traffic to monitor. If you do not specify a traffic direction, the source interface sends both sent and received traffic. <ul style="list-style-type: none"> • both—Monitors both received and sent traffic. • rx—Monitors received traffic. • tx—Monitors sent traffic.
Step 5	<p>monitor session <i>session_number</i> destination remote vlan <i>vlan-id</i></p> <p>Example:</p> <pre>Device(config)# monitor session 1 destination remote vlan 100</pre>	<p>Specifies the RSPAN session, the destination RSPAN VLAN, and the destination-port group.</p> <ul style="list-style-type: none"> • For <i>session_number</i>, enter the number defined in Step 4. • For <i>vlan-id</i>, specify the source RSPAN VLAN to monitor.
Step 6	<p>end</p> <p>Example:</p> <pre>Device(config)# end</pre>	<p>Returns to privileged EXEC mode.</p>
Step 7	<p>show running-config</p> <p>Example:</p> <pre>Device# show running-config</pre>	<p>Verifies your entries.</p>
Step 8	<p>copy running-config startup-config</p> <p>Example:</p> <pre>Device# copy running-config startup-config</pre>	<p>(Optional) Saves your entries in the configuration file.</p>

Specifying VLANs to Filter

Follow these steps to configure the RSPAN source session to limit RSPAN source traffic to specific VLANs.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **no monitor session** {*session_number* | **all** | **local** | **remote**}
4. **monitor session** *session_number* **source interface** *interface-id*
5. **monitor session** *session_number* **filter vlan** *vlan-id* [, | -]
6. **monitor session** *session_number* **destination remote vlan** *vlan-id*
7. **end**
8. **show running-config**
9. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	no monitor session { <i>session_number</i> all local remote } Example: Device(config)# no monitor session 2	Removes any existing SPAN configuration for the session. <ul style="list-style-type: none"> • For <i>session_number</i>, the range is 1 to 66. • all—Removes all SPAN sessions. • local—Removes all local sessions. • remote—Removes all remote SPAN sessions.
Step 4	monitor session <i>session_number</i> source interface <i>interface-id</i> Example: Device(config)# monitor session 2 source interface gigabitethernet1/0/2 rx	Specifies the characteristics of the source port (monitored port) and SPAN session. <ul style="list-style-type: none"> • For <i>session_number</i>, the range is 1 to 66. • For <i>interface-id</i>, specify the source port to monitor. The interface specified must already be configured as a trunk port.
Step 5	monitor session <i>session_number</i> filter vlan <i>vlan-id</i> [, -]	Limits the SPAN source traffic to specific VLANs.

	Command or Action	Purpose
	<p>Example:</p> <pre>Device(config)# monitor session 2 filter vlan 1 - 5 , 9</pre>	<ul style="list-style-type: none"> • For <i>session_number</i>, enter the session number specified in step 4. • For <i>vlan-id</i>, the range is 1 to 4094. • (Optional) , - Use a comma (,) to specify a series of VLANs or use a hyphen (-) to specify a range of VLANs. Enter a space before and after the comma; enter a space before and after the hyphen.
Step 6	<p>monitor session <i>session_number</i> destination remote vlan <i>vlan-id</i></p> <p>Example:</p> <pre>Device(config)# monitor session 2 destination remote vlan 902</pre>	<p>Specifies the RSPAN session and the destination remote VLAN (RSPAN VLAN).</p> <ul style="list-style-type: none"> • For <i>session_number</i>, enter the session number specified in Step 4. • For <i>vlan-id</i>, specify the RSPAN VLAN to carry the monitored traffic to the destination port.
Step 7	<p>end</p> <p>Example:</p> <pre>Device(config)# end</pre>	Returns to privileged EXEC mode.
Step 8	<p>show running-config</p> <p>Example:</p> <pre>Device# show running-config</pre>	Verifies your entries.
Step 9	<p>copy running-config startup-config</p> <p>Example:</p> <pre>Device# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Creating an RSPAN Destination Session

You configure an RSPAN destination session on a different device or device stack; that is, not the device or device stack on which the source session was configured.

Follow these steps to define the RSPAN VLAN on that device, to create an RSPAN destination session, and to specify the source RSPAN VLAN and the destination port.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **vlan** *vlan-id*

4. **remote-span**
5. **exit**
6. **no monitor session** {*session_number* | **all** | **local** | **remote**}
7. **monitor session** *session_number* **source remote vlan** *vlan-id*
8. **monitor session** *session_number* **destination interface** *interface-id*
9. **end**
10. **show running-config**
11. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	vlan <i>vlan-id</i> Example: Device(config)# vlan 901	Specifies the VLAN ID of the RSPAN VLAN created from the source device, and enters VLAN configuration mode. If both devices are participating in VTP and the RSPAN VLAN ID is from 2 to 1005, Steps 3 through 5 are not required because the RSPAN VLAN ID is propagated through the VTP network.
Step 4	remote-span Example: Device(config-vlan)# remote-span	Identifies the VLAN as the RSPAN VLAN.
Step 5	exit Example: Device(config-vlan)# exit	Returns to global configuration mode.
Step 6	no monitor session { <i>session_number</i> all local remote } Example: Device(config)# no monitor session 1	Removes any existing SPAN configuration for the session. <ul style="list-style-type: none"> • For <i>session_number</i>, the range is 1 to 66. • all—Removes all SPAN sessions. • local—Removes all local sessions.

	Command or Action	Purpose
		<ul style="list-style-type: none"> • remote—Removes all remote SPAN sessions.
Step 7	<p>monitor session <i>session_number</i> source remote vlan <i>vlan-id</i></p> <p>Example:</p> <pre>Device(config)# monitor session 1 source remote vlan 901</pre>	<p>Specifies the RSPAN session and the source RSPAN VLAN.</p> <ul style="list-style-type: none"> • For <i>session_number</i>, the range is 1 to 66. • For <i>vlan-id</i>, specify the source RSPAN VLAN to monitor.
Step 8	<p>monitor session <i>session_number</i> destination interface <i>interface-id</i></p> <p>Example:</p> <pre>Device(config)# monitor session 1 destination interface gigabitethernet2/0/1</pre>	<p>Specifies the RSPAN session and the destination interface.</p> <ul style="list-style-type: none"> • For <i>session_number</i>, enter the number defined in Step 7. <p>In an RSPAN destination session, you must use the same session number for the source RSPAN VLAN and the destination port.</p> <ul style="list-style-type: none"> • For <i>interface-id</i>, specify the destination interface. The destination interface must be a physical interface. • Though visible in the command-line help string, encapsulation replicate is not supported for RSPAN. The original VLAN ID is overwritten by the RSPAN VLAN ID, and all packets appear on the destination port as untagged.
Step 9	<p>end</p> <p>Example:</p> <pre>Device(config)# end</pre>	Returns to privileged EXEC mode.
Step 10	<p>show running-config</p> <p>Example:</p> <pre>Device# show running-config</pre>	Verifies your entries.
Step 11	<p>copy running-config startup-config</p> <p>Example:</p> <pre>Device# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Creating an RSPAN Destination Session and Configuring Incoming Traffic

Follow these steps to create an RSPAN destination session, to specify the source RSPAN VLAN and the destination port, and to enable incoming traffic on the destination port for a network security device (such as a Cisco IDS Sensor Appliance).

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **no monitor session** {*session_number* | **all** | **local** | **remote**}
4. **monitor session** *session_number* **source remote vlan** *vlan-id*
5. **monitor session** *session_number* **destination** {**interface** *interface-id* [, | -] [**ingress** {**dot1q vlan** *vlan-id* | **untagged vlan** *vlan-id* | **vlan** *vlan-id*}]}
6. **end**
7. **show running-config**
8. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	no monitor session { <i>session_number</i> all local remote } Example: Device(config)# no monitor session 2	Removes any existing SPAN configuration for the session. <ul style="list-style-type: none"> • For <i>session_number</i>, the range is 1 to 66. • all—Removes all SPAN sessions. • local—Removes all local sessions. • remote—Removes all remote SPAN sessions.
Step 4	monitor session <i>session_number</i> source remote vlan <i>vlan-id</i> Example: Device(config)# monitor session 2 source remote vlan 901	Specifies the RSPAN session and the source RSPAN VLAN. <ul style="list-style-type: none"> • For <i>session_number</i>, the range is 1 to 66. • For <i>vlan-id</i>, specify the source RSPAN VLAN to monitor.

	Command or Action	Purpose
Step 5	<p>monitor session <i>session_number</i> destination {interface <i>interface-id</i> [, -] [ingress {dot1q vlan <i>vlan-id</i> untagged vlan <i>vlan-id</i> vlan <i>vlan-id</i>}]}</p> <p>Example:</p> <pre>Device(config)# monitor session 2 destination interface gigabitethernet1/0/2 ingress vlan 6</pre>	<p>Specifies the SPAN session, the destination port, the packet encapsulation, and the incoming VLAN and encapsulation.</p> <ul style="list-style-type: none"> For <i>session_number</i>, enter the number defined in Step 5. In an RSPAN destination session, you must use the same session number for the source RSPAN VLAN and the destination port. For <i>interface-id</i>, specify the destination interface. The destination interface must be a physical interface. Though visible in the command-line help string, encapsulation replicate is not supported for RSPAN. The original VLAN ID is overwritten by the RSPAN VLAN ID, and all packets appear on the destination port as untagged. (Optional) [, -] Specifies a series or range of interfaces. Enter a space before and after the comma; enter a space before and after the hyphen. Enter ingress with additional keywords to enable forwarding of incoming traffic on the destination port and to specify the encapsulation type: <ul style="list-style-type: none"> dot1q vlan <i>vlan-id</i>—Forwards incoming packets with IEEE 802.1Q encapsulation with the specified VLAN as the default VLAN. untagged vlan <i>vlan-id</i> or vlan <i>vlan-id</i>—Forwards incoming packets with untagged encapsulation type with the specified VLAN as the default VLAN.
Step 6	<p>end</p> <p>Example:</p> <pre>Device(config)# end</pre>	Returns to privileged EXEC mode.
Step 7	<p>show running-config</p> <p>Example:</p> <pre>Device# show running-config</pre>	Verifies your entries.
Step 8	<p>copy running-config startup-config</p> <p>Example:</p>	(Optional) Saves your entries in the configuration file.

	Command or Action	Purpose
	Device# <code>copy running-config startup-config</code>	

Configuring an FSPAN Session

Follow these steps to create a SPAN session, specify the source (monitored) ports or VLANs and the destination (monitoring) ports, and configure FSPAN for the session.

SUMMARY STEPS

1. `enable`
2. `configure terminal`
3. `no monitor session {session_number | all | local | remote}`
4. `monitor session session_number source {interface interface-id | vlan vlan-id} [, | -] [both | rx | tx]`
5. `monitor session session_number destination {interface interface-id [, | -] [encapsulation replicate]}`
6. `monitor session session_number filter {ip | ipv6 | mac} access-group {access-list-number | name}`
7. `end`
8. `show running-config`
9. `copy running-config startup-config`

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> <code>enable</code>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode.
Step 3	no monitor session {session_number all local remote} Example: Device(config)# <code>no monitor session 2</code>	Removes any existing SPAN configuration for the session. <ul style="list-style-type: none"> • For <i>session_number</i>, the range is 1 to 66. • all—Removes all SPAN sessions. • local—Removes all local sessions. • remote—Removes all remote SPAN sessions.
Step 4	monitor session session_number source {interface interface-id vlan vlan-id} [, -] [both rx tx] Example:	Specifies the SPAN session and the source port (monitored port). <ul style="list-style-type: none"> • For <i>session_number</i>, the range is 1 to 66.

	Command or Action	Purpose
	<pre>Device(config)# monitor session 2 source interface gigabitethernet1/0/1</pre>	<ul style="list-style-type: none"> For <i>interface-id</i>, specifies the source port to monitor. Valid interfaces include physical interfaces and port-channel logical interfaces (port-channel port-channel-number). Valid port-channel numbers are 1 to 48. For <i>vlan-id</i>, specify the source VLAN to monitor. The range is 1 to 4094 (excluding the RSPAN VLAN). <p>Note A single session can include multiple sources (ports or VLANs) defined in a series of commands, but you cannot combine source ports and source VLANs in one session.</p> <ul style="list-style-type: none"> (Optional) [<i>, -</i>]—Specifies a series or range of interfaces. Enter a space before and after the comma; enter a space before and after the hyphen. (Optional) [both rx tx]—Specifies the direction of traffic to monitor. If you do not specify a traffic direction, the SPAN monitors both sent and received traffic. <ul style="list-style-type: none"> both—Monitors both sent and received traffic. This is the default. rx—Monitors received traffic. tx—Monitors sent traffic. <p>Note You can use the monitor session session_number source command multiple times to configure multiple source ports.</p>
Step 5	<p>monitor session <i>session_number</i> destination {interface <i>interface-id</i> [<i>, -</i>] [encapsulation replicate]}</p> <p>Example:</p> <pre>Device(config)# monitor session 2 destination interface gigabitethernet1/0/2 encapsulation replicate</pre>	<p>Specifies the SPAN session and the destination port (monitoring port).</p> <ul style="list-style-type: none"> For <i>session_number</i>, specify the session number entered in Step 4. For destination, specify the following parameters: <ul style="list-style-type: none"> For <i>interface-id</i>, specify the destination port. The destination interface must be a physical port; it cannot be an EtherChannel, and it cannot be a VLAN. (Optional) [<i>, -</i>] Specifies a series or range of interfaces. Enter a space before and after the comma; enter a space before and after the hyphen.

	Command or Action	Purpose
		<ul style="list-style-type: none"> (Optional) encapsulation replicate specifies that the destination interface replicates the source interface encapsulation method. If not selected, the default is to send packets in native form (untagged). <p>Note For local SPAN, you must use the same session number for the source and destination interfaces.</p> <p>You can use monitor session <i>session_number</i> destination command multiple times to configure multiple destination ports.</p>
Step 6	<p>monitor session <i>session_number</i> filter {ip ipv6 mac} access-group {<i>access-list-number</i> <i>name</i>}</p> <p>Example:</p> <pre>Device(config)# monitor session 2 filter ipv6 access-group 4</pre>	<p>Specifies the SPAN session, the types of packets to filter, and the ACLs to use in an FSPAN session.</p> <ul style="list-style-type: none"> For <i>session_number</i>, specify the session number entered in Step 4. For <i>access-list-number</i>, specify the ACL number that you want to use to filter traffic. For <i>name</i>, specify the ACL name that you want to use to filter traffic.
Step 7	<p>end</p> <p>Example:</p> <pre>Device(config)# end</pre>	Returns to privileged EXEC mode.
Step 8	<p>show running-config</p> <p>Example:</p> <pre>Device# show running-config</pre>	Verifies your entries.
Step 9	<p>copy running-config startup-config</p> <p>Example:</p> <pre>Device# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Configuring an FRSPAN Session

Follow these steps to start an RSPAN source session, specify the monitored source and the destination RSPAN VLAN, and configure FRSPAN for the session.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **no monitor session** {*session_number* | **all** | **local** | **remote**}
4. **monitor session** *session_number* **source** {**interface** *interface-id* | **vlan** *vlan-id*} [, | -] [**both** | **rx** | **tx**]
5. **monitor session** *session_number* **destination remote vlan** *vlan-id*
6. **vlan** *vlan-id*
7. **remote-span**
8. **exit**
9. **monitor session** *session_number* **filter** {**ip** | **ipv6** | **mac**} **access-group** {*access-list-number* | *name*}
10. **end**
11. **show running-config**
12. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Device> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: <pre>Device# configure terminal</pre>	Enters global configuration mode.
Step 3	no monitor session { <i>session_number</i> all local remote } Example: <pre>Device(config)# no monitor session 2</pre>	Removes any existing SPAN configuration for the session. <ul style="list-style-type: none"> • For <i>session_number</i>, the range is 1 to 66. • all—Removes all SPAN sessions. • local—Removes all local sessions. • remote—Removes all remote SPAN sessions.
Step 4	monitor session <i>session_number</i> source { interface <i>interface-id</i> vlan <i>vlan-id</i> } [, -] [both rx tx] Example: <pre>Device(config)# monitor session 2 source interface gigabitethernet1/0/1</pre>	Specifies the SPAN session and the source port (monitored port). <ul style="list-style-type: none"> • For <i>session_number</i>, the range is 1 to 66. • For <i>interface-id</i>, specifies the source port to monitor. Valid interfaces include physical interfaces and port-channel logical interfaces (port-channel <i>port-channel-number</i>). Valid port-channel numbers are 1 to 48.

	Command or Action	Purpose
		<ul style="list-style-type: none"> For <i>vlan-id</i>, specify the source VLAN to monitor. The range is 1 to 4094 (excluding the RSPAN VLAN). <p>Note A single session can include multiple sources (ports or VLANs) defined in a series of commands, but you cannot combine source ports and source VLANs in one session.</p> <ul style="list-style-type: none"> (Optional) [, -]—Specifies a series or range of interfaces. Enter a space before and after the comma; enter a space before and after the hyphen. (Optional) [both rx tx]—Specifies the direction of traffic to monitor. If you do not specify a traffic direction, the SPAN monitors both sent and received traffic. both—Monitors both sent and received traffic. This is the default. rx—Monitors received traffic. tx—Monitors sent traffic. <p>Note You can use the monitor session <i>session_number</i> source command multiple times to configure multiple source ports.</p>
Step 5	<p>monitor session <i>session_number</i> destination remote vlan <i>vlan-id</i></p> <p>Example:</p> <pre>Device(config)# monitor session 2 destination remote vlan 5</pre>	<p>Specifies the RSPAN session and the destination RSPAN VLAN.</p> <ul style="list-style-type: none"> For <i>session_number</i>, enter the number defined in Step 4. For <i>vlan-id</i>, specify the destination RSPAN VLAN to monitor.
Step 6	<p>vlan <i>vlan-id</i></p> <p>Example:</p> <pre>Device(config)# vlan 10</pre>	<p>Enters the VLAN configuration mode. For <i>vlan-id</i>, specify the source RSPAN VLAN to monitor.</p>
Step 7	<p>remote-span</p> <p>Example:</p> <pre>Device(config-vlan)# remote-span</pre>	<p>Specifies that the VLAN you specified in Step 5 is part of the RSPAN VLAN.</p>
Step 8	<p>exit</p> <p>Example:</p>	<p>Returns to global configuration mode.</p>

	Command or Action	Purpose
	Device(config-vlan)# exit	
Step 9	<p>monitor session <i>session_number</i> filter {ip ipv6 mac}</p> <p>access-group {<i>access-list-number</i> <i>name</i>}</p> <p>Example:</p> <pre>Device(config)# monitor session 2 filter ip access-group 7</pre>	<p>Specifies the RSPAN session, the types of packets to filter, and the ACLs to use in an FRSPAN session.</p> <ul style="list-style-type: none"> • For <i>session_number</i>, specify the session number entered in Step 4. • For <i>access-list-number</i>, specify the ACL number that you want to use to filter traffic. • For <i>name</i>, specify the ACL name that you want to use to filter traffic.
Step 10	<p>end</p> <p>Example:</p> <pre>Device(config)# end</pre>	Returns to privileged EXEC mode.
Step 11	<p>show running-config</p> <p>Example:</p> <pre>Device# show running-config</pre>	Verifies your entries.
Step 12	<p>copy running-config startup-config</p> <p>Example:</p> <pre>Device# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Monitoring SPAN and RSPAN Operations

The following table describes the command used to display SPAN and RSPAN operations configuration and results to monitor operations:

Table 70: Monitoring SPAN and RSPAN Operations

Command	Purpose
show monitor	Displays the current SPAN, RSPAN, FSPAN, or FRSPAN configuration.

SPAN and RSPAN Configuration Examples

Example: Configuring Local SPAN

This example shows how to set up SPAN session 1 for monitoring source port traffic to a destination port. First, any existing SPAN configuration for session 1 is deleted, and then bidirectional traffic is mirrored from source Gigabit Ethernet port 1 to destination Gigabit Ethernet port 2, retaining the encapsulation method.

```
Device> enable
Device# configure terminal
Device(config)# no monitor session 1
Device(config)# monitor session 1 source interface gigabitethernet1/0/1
Device(config)# monitor session 1 destination interface gigabitethernet1/0/2
encapsulation replicate
Device(config)# end
```

This example shows how to remove port 1 as a SPAN source for SPAN session 1:

```
Device> enable
Device# configure terminal
Device(config)# no monitor session 1 source interface gigabitethernet1/0/1
Device(config)# end
```

This example shows how to disable received traffic monitoring on port 1, which was configured for bidirectional monitoring:

```
Device> enable
Device# configure terminal
Device(config)# no monitor session 1 source interface gigabitethernet1/0/1 rx
```

The monitoring of traffic received on port 1 is disabled, but traffic sent from this port continues to be monitored.

This example shows how to remove any existing configuration on SPAN session 2, configure SPAN session 2 to monitor received traffic on all ports belonging to VLANs 1 through 3, and send it to destination Gigabit Ethernet port 2. The configuration is then modified to also monitor all traffic on all ports belonging to VLAN 10.

```
Device> enable
Device# configure terminal
Device(config)# no monitor session 2
Device(config)# monitor session 2 source vlan 1 - 3 rx
Device(config)# monitor session 2 destination interface gigabitethernet1/0/2
Device(config)# monitor session 2 source vlan 10
Device(config)# end
```

This example shows how to remove any existing configuration on SPAN session 2, configure SPAN session 2 to monitor received traffic on Gigabit Ethernet source port 1, and send it to destination Gigabit Ethernet port 2 with the same egress encapsulation type as the source port, and to enable ingress forwarding with VLAN 6 as the default ingress VLAN:

```
Device> enable
Device# configure terminal
Device(config)# no monitor session 2
Device(config)# monitor session 2 source gigabitethernet0/1 rx
```

```
Device(config)# monitor session 2 destination interface gigabitethernet0/2 encapsulation
replicate ingress vlan 6
Device(config)# end
```

This example shows how to remove any existing configuration on SPAN session 2, configure SPAN session 2 to monitor traffic received on Gigabit Ethernet trunk port 2, and send traffic for only VLANs 1 through 5 and VLAN 9 to destination Gigabit Ethernet port 1:

```
Device> enable
Device# configure terminal
Device(config)# no monitor session 2
Device(config)# monitor session 2 source interface gigabitethernet1/0/2 rx
Device(config)# monitor session 2 filter vlan 1 - 5 , 9
Device(config)# monitor session 2 destination interface gigabitethernet1/0/1
Device(config)# end
```

Examples: Creating an RSPAN VLAN

This example shows how to create the RSPAN VLAN 901:

```
Device> enable
Device# configure terminal
Device(config)# vlan 901
Device(config-vlan)# remote span
Device(config-vlan)# end
```

This example shows how to remove any existing RSPAN configuration for session 1, configure RSPAN session 1 to monitor multiple source interfaces, and configure the destination as RSPAN VLAN 901:

```
Device> enable
Device# configure terminal
Device(config)# no monitor session 1
Device(config)# monitor session 1 source interface gigabitethernet1/0/1 tx
Device(config)# monitor session 1 source interface gigabitethernet1/0/2 rx
Device(config)# monitor session 1 source interface port-channel 2
Device(config)# monitor session 1 destination remote vlan 901
Device(config)# end
```

This example shows how to remove any existing configuration on RSPAN session 2, configure RSPAN session 2 to monitor traffic received on trunk port 2, and send traffic for only VLANs 1 through 5 and 9 to destination RSPAN VLAN 902:

```
Device> enable
Device# configure terminal
Device(config)# no monitor session 2
Device(config)# monitor session 2 source interface gigabitethernet1/0/2 rx
Device(config)# monitor session 2 filter vlan 1 - 5 , 9
Device(config)# monitor session 2 destination remote vlan 902
Device(config)# end
```

This example shows how to configure VLAN 901 as the source remote VLAN and port 1 as the destination interface:

```
Device> enable
Device# configure terminal
Device(config)# monitor session 1 source remote vlan 901
```

```
Device(config)# monitor session 1 destination interface gigabitethernet2/0/1
Device(config)# end
```

This example shows how to configure VLAN 901 as the source remote VLAN in RSPAN session 2, to configure Gigabit Ethernet source port 2 as the destination interface, and to enable forwarding of incoming traffic on the interface with VLAN 6 as the default receiving VLAN:

```
Device> enable
Device# configure terminal
Device(config)# monitor session 2 source remote vlan 901
Device(config)# monitor session 2 destination interface gigabitethernet1/0/2 ingress vlan 6
Device(config)# end
```

Additional References

Related Documents

Related Topic	Document Title
System Commands	<i>Network Management Command Reference, Cisco IOS Release 15.2(2)E</i>

Error Message Decoder

Description	Link
To help you research and resolve system error messages in this release, use the Error Message Decoder tool.	https://www.cisco.com/cgi-bin/Support/Errordecoder/index.cgi

Standards and RFCs

Standard/RFC	Title
None	-

MIBs

MIB	MIBs Link
All the supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
<p>The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies.</p> <p>To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds.</p> <p>Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.</p>	<p>http://www.cisco.com/support</p>

Feature History and Information for SPAN and RSPAN

Release	Modification
Cisco IOS Release 15.0(2)EX1	<p>Switch Port Analyzer (SPAN): Allows monitoring of device traffic on a port or VLAN using a sniffer/analyzer or RMON probe.</p> <p>This feature was introduced.</p>
Cisco IOS 15.0(2)EX1	<p>Flow-based Switch Port Analyzer (SPAN): Provides a method to capture only required data between end hosts by using specified filters. The filters are defined in terms of access lists that limit IPv4, IPv6 or IPv4 + IPv6, or non-IP traffic (MAC) between specified source and destination addresses.</p> <p>This feature was introduced.</p>
Cisco IOS Release 15.0(2)EX1	<p>SPAN destination port support on EtherChannels: Provides the ability to configure a SPAN destination port on an EtherChannel.</p> <p>This feature was introduced.</p>

Release	Modification
Cisco IOS Release 15.0(2)EX1	<p>Switch Port Analyzer (SPAN) - distributed egress SPAN: Provides distributed egress SPAN functionality onto line cards in conjunction with ingress SPAN already been distributed to line cards. By distributing egress SPAN functionalities onto line cards, the performance of the system is improved.</p> <p>This feature was introduced.</p>



PART **VII**

Cisco Flexible NetFlow

- [Configuring Flexible NetFlow, on page 683](#)



CHAPTER 38

Configuring Flexible NetFlow

- [Prerequisites for Flexible NetFlow, on page 683](#)
- [Restrictions for Flexible NetFlow, on page 684](#)
- [Information About Flexible Netflow, on page 686](#)
- [How to Configure Flexible Netflow, on page 690](#)
- [Monitoring Flexible NetFlow, on page 703](#)
- [Configuration Examples for Flexible NetFlow, on page 704](#)
- [Additional References for NetFlow, on page 704](#)
- [Feature Information for Flexible NetFlow, on page 705](#)

Prerequisites for Flexible NetFlow

- Flexible NetFlow is supported on the Catalyst 2960-X Switch and the Catalyst 2960-XR Switch with a Cisco ONE for Access license. Catalyst 2960-XR is not stackable with the Catalyst 2960-X platform.
- One of the following must be enabled on your device and on any interfaces on which you want to enable Flexible NetFlow: Cisco Express Forwarding or distributed Cisco Express Forwarding.
- The targets for attaching a NetFlow monitor are the following:
 - **Port**—Monitor attachment is only supported on physical interfaces and not on logical interfaces, such as EtherChannels. The physical interface could be a routed port or a switched port.
 - **VLAN**—Monitor attachment is supported on VLAN interfaces only (SVI) and not on a Layer 2 VLAN.
- You are familiar with the Flexible NetFlow key fields as they are defined in the following commands:
 - **match datalink**—Datalink (layer2) fields
 - **match ipv4**—IPv4 fields
 - **match ipv6**—IPv6 fields
 - **match transport**—Transport layer fields
- You are familiar with the Flexible NetFlow non key fields as they are defined in the following commands:
 - **collect counter**—Counter fields

- **collect flow**—Flow identifying fields
- **collect interface**—Interface fields
- **collect timestamp**—Timestamp fields
- **collect transport**—Transport layer fields

Restrictions for Flexible NetFlow

The following restrictions apply to Flexible NetFlow and Flexible NetFlow Lite:

General Restrictions:

- InterSwitch Link (ISL) is not supported.
- Policy-based NetFlow is not supported.
- Cisco TrustSec monitoring is not supported.
- Access control lists (ACL)-based NetFlow is not supported.
- Only NetFlow Version 9 is supported for Flexible NetFlow exporter using the *export-protocol* command option.
- NetFlow Version 5 is not supported.

Flow Record Restrictions:

- When a flow monitor has configured the **collect interface output** command as the collect field in the flow record, the field will return a value of **NULL** when a flow gets created for any of the following addresses:
 - L2 broadcast and multicast
 - L3 broadcast and multicast
 - L2 unknown destination.

When a flow monitor has the collect interface output configured as the collect field in the flow record, the output interface is detected based on the destination IP address on the device. For the different flow monitors, you must configure the following commands:

- IPv4 flow monitor--Configure the **match ipv4 destination address** command.
- IPv6 flow monitor--Configure the **match ipv6 destination address** command.
- Datalink flow monitor--Configure the **match datalink mac destination address input** command.
- Predefined flow records are not supported.

Monitor Restrictions:

- Monitor attachment is only supported in the ingress direction.
- One monitor per interface is supported, although multiple exporters per interface are supported.

- Only permanent and normal cache is supported for the monitor; immediate cache is not supported.
- Changing any monitor parameter will not be supported when it is applied on any of the interfaces or VLANs.
- When both the port and VLANs have monitors attached, then VLAN monitor will overwrite the port monitor for traffic coming on the port.
- Flow monitor type and traffic type (type means IPv4, IPv6, and data link) should be same for the flows to be created.
- You cannot attach an IP and a port-based monitor to an interface. A 48-port device supports a maximum of 48 monitors (IP or port-based) and for 256 SVIs, you can configure up to 256 monitors (IP or port-based).
- When running the **show flow monitor** *flow_name* **cache** command, the device displays cache information from an earlier switch software version (Catalyst 2960-S) with all fields entered as zero. Ignore these fields, as they are inapplicable to the switch.

Sampler Restrictions:

- For both port and VLANs, a total of only 4 samplers (random or deterministic) are supported on the device.
- The sampling minimum rate for both modes is 1 out of 32 flows, and the sampling maximum rate for both modes is 1 out of 1022 flows.
- Use the **ip flow monitor** *monitor_name* sampler *sampler_name* **input** command to associate a sampler with a monitor while attaching it to an interface.
- When you attach a monitor using a deterministic sampler, every attachment with the same sampler uses one new free sampler from the switch (hardware) out of the 4 available samplers. You are not allowed to attach a monitor with any sampler, beyond 4 attachments.

When you attach a monitor using a random sampler, only the first attachment uses a new sampler from the switch (hardware). The remainder of all of the attachments using the same sampler, share the same sampler.

Because of this behavior, when using a deterministic sampler, you can always make sure that the correct number of flows are sampled by comparing the sampling rate and what the device sends. If the same random sampler is used with multiple interfaces, flows from any interface can always be sampled, and flows from other interfaces can always be skipped.

Stacking Restrictions:

- Each device in a stack (hardware) can support the creation of a maximum of 16,000 flows at any time. But as the flows are periodically pushed to the software cache, the software cache can hold a much larger amount of flows (1048 Kb flows). From the hardware flow cache, every 20 seconds (termed as poll timer), 200 flows (termed as poll entries) are pushed to software.
 - Use the **remote command all show platform hulf-fnf poll** command to report on the current NetFlow polling parameters of each switch.
 - Use the **show platform hulf-fnf poll** command to report on the current NetFlow polling parameters of the active switch.
- Network flows and statistics are collected at the line rate.

Information About Flexible Netflow

Flexible NetFlow Overview

Flexible NetFlow uses flows to provide statistics for accounting, network monitoring, and network planning.

A flow is a unidirectional stream of packets that arrives on a source interface and has the same values for the keys. A key is an identified value for a field within the packet. You create a flow using a flow record to define the unique keys for your flow.

The device supports the Flexible NetFlow feature that enables enhanced network anomalies and security detection. Flexible NetFlow allows you to define an optimal flow record for a particular application by selecting the keys from a large collection of predefined fields.

All key values must match for the packet to count in a given flow. A flow might gather other fields of interest, depending on the export record version that you configure. Flows are stored in the Flexible NetFlow cache.

You can export the data that Flexible NetFlow gathers for your flow by using an exporter and export this data to a remote system such as a Flexible NetFlow collector. The Flexible NetFlow collector can use an IPv4 address.

You define the size of the data that you want to collect for a flow using a monitor. The monitor combines the flow record and exporter with the Flexible NetFlow cache information.

Flexible NetFlow Components

Flexible NetFlow consists of components that can be used together in several variations to perform traffic analysis and data export. The user-defined flow records and the component structure of Flexible NetFlow facilitates the creation of various configurations for traffic analysis and data export on a networking device with a minimum number of configuration commands. Each flow monitor can have a unique combination of flow record, flow exporter, and cache type. If you change a parameter such as the destination IP address for a flow exporter, it is automatically changed for all the flow monitors that use the flow exporter. The same flow monitor can be used in conjunction with different flow samplers to sample the same type of network traffic at different rates on different interfaces. The following sections provide more information on Flexible NetFlow components:

Flow Records

In Flexible NetFlow a combination of key and nonkey fields is called a record. Flexible NetFlow records are assigned to Flexible NetFlow flow monitors to define the cache that is used for storing flow data.

A flow record defines the keys that Flexible NetFlow uses to identify packets in the flow, as well as other fields of interest that Flexible NetFlow gathers for the flow. You can define a flow record with any combination of keys and fields of interest. The device supports a rich set of keys. A flow record also defines the types of counters gathered per flow. You can configure 64-bit packet or byte counters. The device enables the following match fields as the defaults when you create a flow record:

- match datalink—Layer 2 attributes
- match ipv4—IPv4 attributes
- match ipv6—IPv6 attributes

- match transport—Transport layer fields
- match wireless—Wireless fields

User-Defined Records

Flexible NetFlow enables you to define your own records for a Flexible NetFlow flow monitor cache by specifying the key and nonkey fields to customize the data collection to your specific requirements. When you define your own records for a Flexible NetFlow flow monitor cache, they are referred to as *user-defined records*. The values in nonkey fields are added to flows to provide additional information about the traffic in the flows. A change in the value of a nonkey field does not create a new flow. In most cases the values for nonkey fields are taken from only the first packet in the flow. Flexible NetFlow enables you to capture counter values such as the number of bytes and packets in a flow as nonkey fields.

Flexible NetFlow adds a new Version 9 export format field type for the header and packet section types. Flexible NetFlow will communicate to the NetFlow collector the configured section sizes in the corresponding Version 9 export template fields. The payload sections will have a corresponding length field that can be used to collect the actual size of the collected section.

Flow Exporters

Flow exporters export the data in the flow monitor cache to a remote system, such as a server running NetFlow collector, for analysis and storage. Flow exporters are created as separate entities in the configuration. Flow exporters are assigned to flow monitors to provide data export capability for the flow monitors. You can create several flow exporters and assign them to one or more flow monitors to provide several export destinations. You can create one flow exporter and apply it to several flow monitors.

NetFlow Data Export Format Version 9

The basic output of NetFlow is a flow record. Several different formats for flow records have evolved as NetFlow has matured. The most recent evolution of the NetFlow export format is known as Version 9. The distinguishing feature of the NetFlow Version 9 export format is that it is template-based. Templates provide an extensible design to the record format, a feature that should allow future enhancements to NetFlow services without requiring concurrent changes to the basic flow-record format. Using templates provides several key benefits:

- Third-party business partners who produce applications that provide collector or display services for NetFlow do not have to recompile their applications each time a new NetFlow feature is added. Instead, they should be able to use an external data file that documents the known template formats.
- New features can be added to NetFlow quickly without breaking current implementations.
- NetFlow is “future-proofed” against new or developing protocols because the Version 9 format can be adapted to provide support for them.

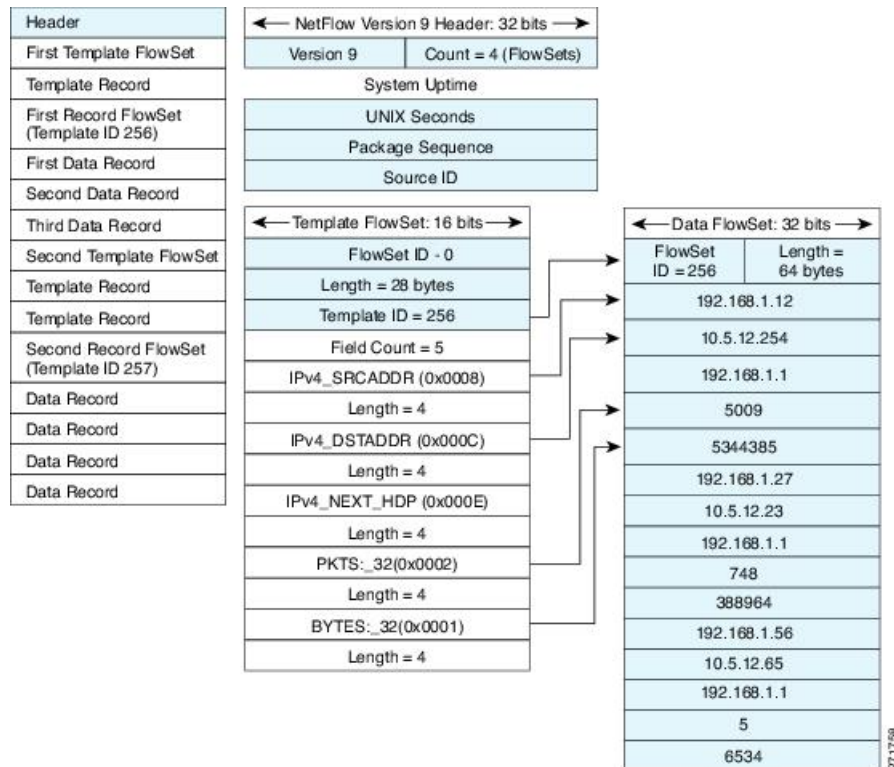
The Version 9 export format consists of a packet header followed by one or more template flow or data flow sets. A template flow set provides a description of the fields that will be present in future data flow sets. These data flow sets may occur later within the same export packet or in subsequent export packets. Template flow and data flow sets can be intermingled within a single export packet, as illustrated in the figure below.

Figure 66: Version 9 Export Packet



NetFlow Version 9 will periodically export the template data so the NetFlow collector will understand what data is to be sent and also export the data flow set for the template. The key advantage to Flexible NetFlow is that the user configures a flow record, which is effectively converted to a Version 9 template and then forwarded to the collector. The figure below is a detailed example of the NetFlow Version 9 export format, including the header, template flow, and data flow sets.

Figure 67: Detailed Example of the NetFlow Version 9 Export Format



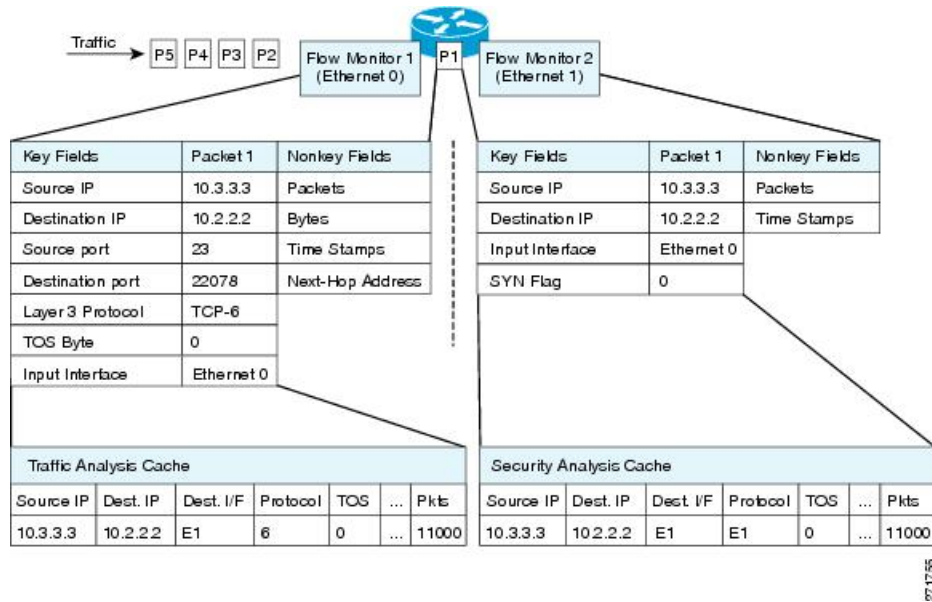
Flow Monitors

Flow monitors are the Flexible NetFlow component that is applied to interfaces to perform network traffic monitoring.

Flow data is collected from the network traffic and added to the flow monitor cache during the monitoring process based on the key and nonkey fields in the flow record.

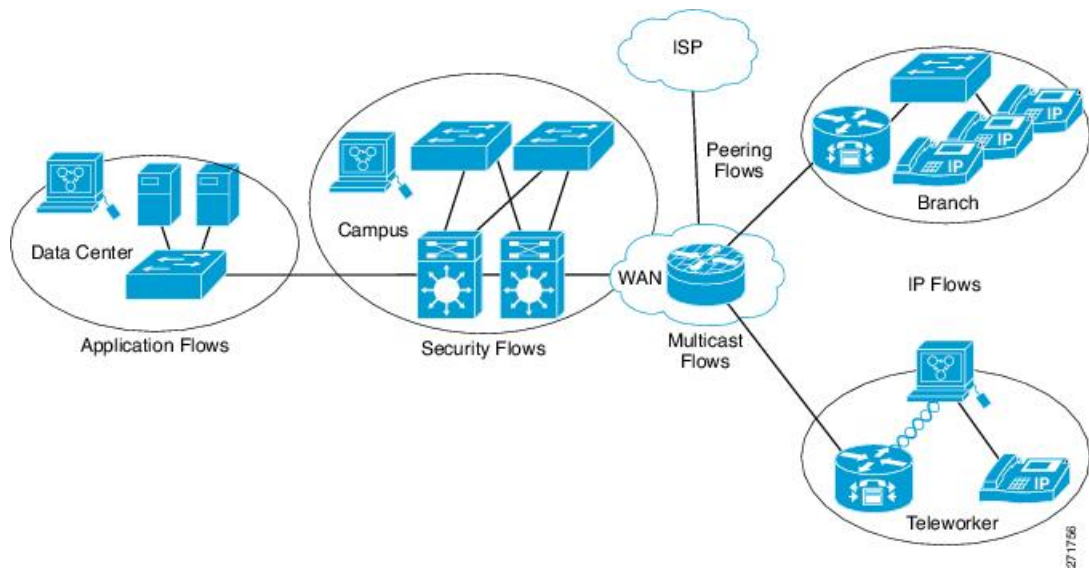
Flexible NetFlow can be used to perform different types of analysis on the same traffic. In the figure below, packet 1 is analyzed using a record designed for standard traffic analysis on the input interface and a record designed for security analysis on the output interface.

Figure 68: Example of Using Two Flow Monitors to Analyze the Same Traffic



The figure below shows a more complex example of how you can apply different types of flow monitors with custom records.

Figure 69: Complex Example of Using Multiple Types of Flow Monitors with Custom Records



Normal

The default cache type is “normal”. In this mode, the entries in the cache are aged out according to the timeout active and timeout inactive settings. When a cache entry is aged out, it is removed from the cache and exported via any exporters configured.

Flow Samplers

Flow samplers are created as separate components in a router's configuration. Flow samplers are used to reduce the load on the device that is running Flexible NetFlow by limiting the number of packets that are selected for analysis.

Samplers use random sampling techniques (modes); that is, a randomly selected sampling position is used each time a sample is taken.

Flow sampling exchanges monitoring accuracy for router performance. When you apply a sampler to a flow monitor, the overhead load on the router of running the flow monitor is reduced because the number of packets that the flow monitor must analyze is reduced. The reduction in the number of packets that are analyzed by the flow monitor causes a corresponding reduction in the accuracy of the information stored in the flow monitor's cache.

Samplers are combined with flow monitors when they are applied to an interface with the **ip flow monitor** command.

Default Settings

The following table lists the Flexible NetFlow default settings for the device.

Table 71: Default Flexible NetFlow Settings

Setting	Default
Flow active timeout	1800 seconds Note The default value for this setting may be too high for your specific Flexible NetFlow configuration. You may want to consider changing it to a lower value of 180 or 300 seconds.
Flow timeout inactive	Enabled, 30 seconds
Flow update timeout	1800 seconds
Default cache size	16640 entries

In Cisco IOS Release 15.2(5)E1, Flexible NetFlow polling was changed from 200 entries every 20 seconds to 2000 entries every 5 seconds. Based on this change, the current flow count will reflect the actual hardware flow count, and continuously active flows will experience active timeout. All flows will be exported as per the configured timeout values.

How to Configure Flexible Netflow

To configure Flexible Netflow, follow these general steps:

1. Create a flow record by specifying keys and non-key fields to the flow.
2. Create an optional flow exporter by specifying the protocol and transport destination port, destination, and other parameters.

3. Create a flow monitor based on the flow record and flow exporter.
4. Create an optional sampler.
5. Apply the flow monitor to a Layer 2 port, Layer 3 port, or VLAN.

Creating a Flow Record

Perform this task to configure a customized flow record.

Customized flow records are used to analyze traffic data for a specific purpose. A customized flow record must have at least one **match** criterion for use as the key field and typically has at least one **collect** criterion for use as a nonkey field.

There are hundreds of possible permutations of customized flow records. This task shows the steps that are used to create one of the possible permutations. Modify the steps in this task as appropriate to create a customized flow record for your requirements.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **flow record** *record-name*
4. **description** *description*
5. **match** {ip | ipv6} {destination | source} address
6. Repeat Step 5 as required to configure additional key fields for the record.
7. **match flow cts** {source | destination} group-tag
8.
 - **collect counter** {bytes [exported | long] | flows [exported] | packets} [exported | long]
 - or
 - **collect timestamp sys-uptime** {first | last}
9. Repeat the above step as required to configure additional nonkey fields for the record.
10. **end**
11. **show flow record** *record-name*
12. **show running-config flow record** *record-name*

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.

	Command or Action	Purpose
Step 3	<p>flow record <i>record-name</i></p> <p>Example:</p> <pre>Device(config)# flow record FLOW-RECORD-1</pre>	<p>Creates a flow record and enters Flexible NetFlow flow record configuration mode.</p> <ul style="list-style-type: none"> This command also allows you to modify an existing flow record.
Step 4	<p>description <i>description</i></p> <p>Example:</p> <pre>Device(config-flow-record)# description Used for basic traffic analysis</pre>	(Optional) Creates a description for the flow record.
Step 5	<p>match {ip ipv6} {destination source} address</p> <p>Example:</p> <pre>Device(config-flow-record)# match ipv4 destination address</pre>	<p>Note This example configures the IPv4 destination address as a key field for the record. For information about the other key fields available for the match ipv4 command, and the other match commands that are available to configure key fields.</p>
Step 6	Repeat Step 5 as required to configure additional key fields for the record.	—
Step 7	<p>match flow cts {source destination} group-tag</p> <p>Example:</p> <pre>Device(config-flow-record)# match flow cts source group-tag</pre> <pre>Device(config-flow-record)# match flow cts destination group-tag</pre>	<p>Note This example configures the CTS source group tag and destination group tag as a key field for the record. For information about the other key fields available for the match ipv4/ipv6 command, and the other match commands that are available to configure key fields.</p>

	Command or Action	Purpose
		<p>Note</p> <ul style="list-style-type: none"> • Ingress: <ul style="list-style-type: none"> • In an incoming packet, if a header is present, SGT will reflect the same value as the header. If no value is present, it will show zero. • The DGT value will not depend on the ingress port SGACL configuration. • Egress: <ul style="list-style-type: none"> • If either propagate SGT or CTS is disabled on the egress interface, then SGT will be zero. • In an outgoing packet, if SGACL configuration that corresponds to the (SGT, DGT) exists, DGT will be non-zero. • If SGACL is disabled on the egress port/VLAN or if global SGACL enforcement is disabled, then DGT will be zero
Step 8	<ul style="list-style-type: none"> • collect counter {bytes [exported long] flows [exported] packets} [exported long] • or collect timestamp sys-uptime {first last} <p>Example:</p> <pre>Device(config-flow-record)# collect counter bytes</pre>	<p>Configures the input interface as a nonkey field for the record.</p> <p>Note This example configures the input interface as a nonkey field for the record.</p>
Step 9	Repeat the above step as required to configure additional nonkey fields for the record.	—
Step 10	<p>end</p> <p>Example:</p> <pre>Device(config-flow-record)# end</pre>	Exits Flexible NetFlow flow record configuration mode and returns to privileged EXEC mode.
Step 11	<p>show flow record <i>record-name</i></p> <p>Example:</p> <pre>Device# show flow record FLOW_RECORD-1</pre>	(Optional) Displays the current status of the specified flow record.

	Command or Action	Purpose
Step 12	show running-config flow record <i>record-name</i> Example: <pre>Device# show running-config flow record FLOW_RECORD-1</pre>	(Optional) Displays the configuration of the specified flow record.

Creating a Flow Exporter

You can create a flow export to define the export parameters for a flow.



Note Each flow exporter supports only one destination. If you want to export the data to multiple destinations, you must configure multiple flow exporters and assign them to the flow monitor.

You can export to a destination using IPv4 address.

SUMMARY STEPS

1. **configure terminal**
2. **flow exporter** *name*
3. **description** *string*
4. **destination** {*ipv4-address*} [**vrf** *vrf-name*]
5. **dscp** *value*
6. **source** { *source type* }
7. **transport udp** *number*
8. **ttl** *seconds*
9. **export-protocol** {*netflow-v9*}
10. **end**
11. **show flow exporter** [**name** *record-name*]
12. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>Device# configure terminal</pre>	Enters global configuration mode.
Step 2	flow exporter <i>name</i> Example: <pre>Device(config)# flow exporter ExportTest</pre>	Creates a flow exporter and enters flow exporter configuration mode.

	Command or Action	Purpose
Step 3	description <i>string</i> Example: Device(config-flow-exporter)# description ExportV9	(Optional) Describes this flow record as a maximum 63-character string.
Step 4	destination { <i>ipv4-address</i> } [vrf <i>vrf-name</i>] Example: Device(config-flow-exporter)# destination 192.0.2.1 (IPv4 destination)	Sets the IPv4 destination address or hostname for this exporter.
Step 5	dscp <i>value</i> Example: Device(config-flow-exporter)# dscp 0	(Optional) Specifies the differentiated services codepoint value. The range is from 0 to 63. The default is 0.
Step 6	source { <i>source type</i> } Example: Device(config-flow-exporter)# source gigabitEthernet1/0/1	(Optional) Specifies the interface to use to reach the NetFlow collector at the configured destination. The following interfaces can be configured as source: \
Step 7	transport udp <i>number</i> Example: Device(config-flow-exporter)# transport udp 200	(Optional) Specifies the UDP port to use to reach the NetFlow collector. The range is from 1 to 65536
Step 8	ttl <i>seconds</i> Example: Device(config-flow-exporter)# ttl 210	(Optional) Configures the time-to-live (TTL) value for datagrams sent by the exporter. The range is from 1 to 255 seconds. The default is 255.
Step 9	export-protocol { <i>netflow-v9</i> } Example: Device(config-flow-exporter)# export-protocol netflow-v9	Specifies the version of the NetFlow export protocol used by the exporter.
Step 10	end Example:	Returns to privileged EXEC mode.

	Command or Action	Purpose
	Device (config-flow-record) # end	
Step 11	show flow exporter [<i>name record-name</i>] Example: Device# show flow exporter ExportTest	(Optional) Displays information about NetFlow flow exporters.
Step 12	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

What to do next

Define a flow monitor based on the flow record and flow exporter.

Creating a Flow Monitor

Perform this required task to create a customized flow monitor.

Each flow monitor has a separate cache assigned to it. Each flow monitor requires a record to define the contents and layout of its cache entries. These record formats can be a user-defined format. An advanced user can create a customized format using the **flow record** command.

Before you begin

If you want to use a customized record, you must create the customized record before you can perform this task. If you want to add a flow exporter to the flow monitor for data export, you must create the exporter before you can complete this task.



Note You must use the **no ip flow monitor** command to remove a flow monitor from all of the interfaces to which you have applied it before you can modify the parameters for the **record** command on the flow monitor.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **flow monitor** *monitor-name*
4. **description** *description*
5. **record** {*record-name*}
6. **cache** {**entries** *number* | **timeout** {**active** | **inactive** | **update**} *seconds* | { **normal** }
7. Repeat Step 6 as required to finish modifying the cache parameters for this flow monitor.

8. **exporter** *exporter-name*
9. **end**
10. **show flow monitor** [[**name**] *monitor-name* [**cache** [**format** {**csv** | **record** | **table**}]]]
11. **show running-config flow monitor** *monitor-name*
12. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	flow monitor <i>monitor-name</i> Example: Device(config)# flow monitor FLOW-MONITOR-1	Creates a flow monitor and enters Flexible NetFlow flow monitor configuration mode. <ul style="list-style-type: none"> • This command also allows you to modify an existing flow monitor.
Step 4	description <i>description</i> Example: Device(config-flow-monitor)# description Used for basic ipv4 traffic analysis	(Optional) Creates a description for the flow monitor.
Step 5	record { <i>record-name</i> } Example: Device(config-flow-monitor)# record FLOW-RECORD-1	Specifies the record for the flow monitor.
Step 6	cache { entries <i>number</i> timeout { active inactive update } <i>seconds</i> { normal } Example:	(Optional) Modifies the flow monitor cache parameters such as timeout values, number of cache entries, and the cache type. <ul style="list-style-type: none"> • timeout active <i>seconds</i>—Configure the active flow timeout. This defines the granularity of the traffic analysis. The range is from 1 to 604800 seconds. The default is 1800. Typical values are 60 or 300 seconds. See the Configuring Data Export for Cisco IOS Flexible NetFlow with Flow Exporters document for recommended values. <p>Note Although visible in the command line help, the entries keyword and inactive and update timeouts are not supported.</p>

	Command or Action	Purpose
Step 7	Repeat Step 6 as required to finish modifying the cache parameters for this flow monitor.	—
Step 8	exporter <i>exporter-name</i> Example: Device(config-flow-monitor)# exporter EXPORTER-1	(Optional) Specifies the name of an exporter that was created previously.
Step 9	end Example: Device(config-flow-monitor)# end	Exits Flexible NetFlow flow monitor configuration mode and returns to privileged EXEC mode.
Step 10	show flow monitor [[name] <i>monitor-name</i> [cache [format { csv record table }]]] Example: Device# show flow monitor FLOW-MONITOR-2 cache	(Optional) Displays the status for a Flexible NetFlow flow monitor.
Step 11	show running-config flow monitor <i>monitor-name</i> Example: Device# show running-config flow monitor FLOW_MONITOR-1	(Optional) Displays the configuration of the specified flow monitor.
Step 12	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Creating a Sampler

You can create a sampler to define the NetFlow sampling rate for a flow.

SUMMARY STEPS

1. **configure terminal**
2. **sampler** *name*
3. **description** *string*
4. **mode** {**deterministic** {*m - n*} | **random** {*m - n*}}
5. **end**
6. **show sampler** [*name*]
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 2	sampler <i>name</i> Example: Device(config)# sampler SampleTest	Creates a sampler and enters flow sampler configuration mode.
Step 3	description <i>string</i> Example: Device(config-flow-sampler)# description samples	(Optional) Describes this flow record as a maximum 63-character string.
Step 4	mode {deterministic {<i>m</i> - <i>n</i>} random {<i>m</i> - <i>n</i>}} Example: Device(config-flow-sampler)# mode random 1 out-of-1022	Defines the random sample mode. You can configure either a random or deterministic sampler to an interface. Select <i>m</i> packets out of an <i>n</i> packet window. The window size to select packets from ranges from 32 to 1022. Note the following when configuring a sampler to an interface: <ul style="list-style-type: none"> • When you attach a monitor using deterministic sampler (for example, s1), every attachment with same sampler s1 uses one new free sampler from the device (hardware) out of 4 available samplers. Therefore, beyond 4 attachments, you are not allowed to attach a monitor with any sampler. • In contrast, when you attach a monitor using random sampler (for example-again, s1), only the first attachment uses a new sampler from the device (hardware). The rest of all attachments using the same sampler s1, share the same sampler. • Due to this behavior, when using a deterministic sampler, you can always make sure the correct number of flows are sampled by comparing the sampling rate and what the device sends. If the same random sampler is used with multiple interfaces, flows from an interface can always be sampled, and the flows from other interfaces could be always skipped.

	Command or Action	Purpose
Step 5	end Example: <pre>Device(config-flow-sampler)# end</pre>	Returns to privileged EXEC mode.
Step 6	show sampler [<i>name</i>] Example: <pre>Device show sample SampleTest</pre>	(Optional) Displays information about NetFlow samplers.
Step 7	copy running-config startup-config Example: <pre>Device# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

What to do next

Apply the flow monitor to a source interface or a VLAN.

Applying a Flow to an Interface

You can apply a flow monitor and an optional sampler to an interface.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *type*
3. **{ip flow monitor | ipv6 flow monitor}** *name* [**sampler** *name*] **{input | output}**
4. **end**
5. **show flow interface** [*interface-type number*]
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>Device# configure terminal</pre>	Enters global configuration mode.
Step 2	interface <i>type</i> Example:	Enters interface configuration mode and configures an interface.

	Command or Action	Purpose
	<pre>Device(config)# interface GigabitEthernet1/0/1</pre>	<p>Command parameters for the interface configuration include:</p> <p>Flexible Net Flow is supported only on the service module 1-Gigabit or 10-Gigabit Ethernet interfaces.</p> <p>You cannot attach a NetFlow monitor to a port channel interface. If both service module interfaces are part of an EtherChannel, you should attach the monitor to both physical interfaces.</p>
Step 3	<p>{ip flow monitor ipv6 flow monitor}name [sampler name] {input output}</p> <p>Example:</p> <pre>Device(config-if)# ip flow monitor MonitorTest input</pre>	<p>Associate an IPv4 or an IPv6 flow monitor, and an optional sampler to the interface for input or output packets.</p> <p>You can associate multiple monitors to an interface in both input and output directions.</p> <p>To monitor datalink L2 traffic flows, you would use datalink flow monitor name sampler sampler-name {input} interface command. This specific command associates a datalink L2 flow monitor and required sampler to the interface for input packets. When a datalink flow monitor is assigned to an interface or VLAN record, it only creates flows for non-IPv6 or non-IPv4 traffic.</p>
Step 4	<p>end</p> <p>Example:</p> <pre>Device(config-flow-monitor)# end</pre>	<p>Returns to privileged EXEC mode.</p>
Step 5	<p>show flow interface [interface-type number]</p> <p>Example:</p> <pre>Device# show flow interface</pre>	<p>(Optional) Displays information about NetFlow on an interface.</p>
Step 6	<p>copy running-config startup-config</p> <p>Example:</p> <pre>Device# copy running-config startup-config</pre>	<p>(Optional) Saves your entries in the configuration file.</p>

Configuring Layer 2 NetFlow

You can define Layer 2 keys in Flexible NetFlow records that you can use to capture flows in Layer 2 interfaces.

SUMMARY STEPS

1. configure terminal

2. **flow record** *name*
3. **match datalink** {ethertype | mac {destination {address input} | source {address input}}}
4. **match** { ipv4 {destination | protocol | source | tos} | ipv6 {destination | flow-label | protocol | source | traffic-class} | transport {destination-port | source-port}}
5. **end**
6. **show flow record** [*name*]
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 2	flow record <i>name</i> Example: Device(config)# flow record L2_record Device(config-flow-record)#	Enters flow record configuration mode.
Step 3	match datalink {ethertype mac {destination {address input} source {address input}}} Example: Device(config-flow-record)# match datalink mac source address input Device(config-flow-record)# match datalink mac destination address input	Specifies the Layer 2 attribute as a key. In this example, the keys are the source and destination MAC addresses from the packet at input. Note When a datalink flow monitor is assigned to an interface or VLAN record, it only creates flows for non-IPv4 or non-IPv6 traffic.
Step 4	match { ipv4 {destination protocol source tos} ipv6 {destination flow-label protocol source traffic-class} transport {destination-port source-port}} Example: Device(config-flow-record)# match ipv4 protocol Device(config-flow-record)# match ipv4 tos	Specifies additional Layer 2 attributes as a key. In this example, the keys are IPv4 protocol and ToS.
Step 5	end Example: Device(config-flow-record)# end	Returns to privileged EXEC mode.

	Command or Action	Purpose
Step 6	show flow record [<i>name</i>] Example: Device# <code>show flow record</code>	(Optional) Displays information about NetFlow on an interface.
Step 7	copy running-config startup-config Example: Device# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Monitoring Flexible NetFlow

The commands in the following table can be used to monitor Flexible NetFlow.

Table 72: Flexible NetFlow Monitoring Commands

Command	Purpose
show flow exporter [broker export-ids name <i>name</i> statistics templates]	Displays information about NetFlow flow exporters and statistics.
show flow exporter [name <i>exporter-name</i>]	Displays information about NetFlow flow exporters and statistics.
show flow interface	Displays information about NetFlow interfaces.
show flow monitor [name <i>monitor-name</i>]	Displays information about NetFlow flow monitors and statistics.
show flow monitor statistics	Displays the statistics for the flow monitor
show flow monitor <i>monitor-name</i> cache format { table record csv }	Displays the contents of the cache for the flow monitor, in the format specified.
show flow record [name <i>record-name</i>]	Displays information about NetFlow flow records.
show sampler [broker name <i>name</i>]	Displays information about NetFlow samplers.
show wlan <i>wlan-name</i>	Displays the WLAN configured on the device.

Configuration Examples for Flexible NetFlow

Example: Configuring a Flow

This example shows how to create a flow and apply it to an interface:

```

Device# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.

Device(config)# flow export export1
Device(config-flow-exporter)# destination 10.0.101.254
Device(config-flow-exporter)# transport udp 2055
Device(config-flow-exporter)# exit
Device(config)# flow record record1
Device(config-flow-record)# match ipv4 source address
Device(config-flow-record)# match ipv4 destination address
Device(config-flow-record)# match ipv4 protocol
Device(config-flow-record)# match transport source-port
Device(config-flow-record)# match transport destination-port

Device(config-flow-record)# collect counter byte long
Device(config-flow-record)# collect counter packet long
Device(config-flow-record)# collect timestamp absolute first
Device(config-flow-record)# collect timestamp absolute last
Device(config-flow-record)# exit
Device(config)# flow monitor monitor1
Device(config-flow-monitor)# record record1
Device(config-flow-monitor)# exporter export1
Device(config-flow-monitor)# exit
Device(config)# interface tenGigabitEthernet 1/0/1
Device(config-if)# ip flow monitor monitor1 input
Device(config-if)# end

```

Additional References for NetFlow

Related Documents

Related Topic	Document Title
Flexible NetFlow CLI Commands	NetFlow Command Reference
Catalyst 2960-X commands	Consolidated Platform Command Reference
Catalyst 2960-XR commands	Consolidated Platform Command Reference

Standards and RFCs

Standard/RFC	Title
RFC 3954	Cisco Systems NetFlow Services Export Version 9

MIBs

MB	MIBs Link
	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
<p>The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies.</p> <p>To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds.</p> <p>Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.</p>	http://www.cisco.com/support

Feature Information for Flexible NetFlow

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Table 73: Feature Information for Flexible NetFlow

Feature Name	Releases	Feature Information
Flexible NetFlow	Cisco IOS Release 15.2(5)E1	<p>NetFlow is a Cisco IOS technology that provides statistics on packets flowing through the router. NetFlow is the standard for acquiring IP operational data from IP networks. NetFlow provides data to enable network and security monitoring, network planning, traffic analysis, and IP accounting.</p> <p>In Cisco IOS Release 15.2(5)E1, this feature was introduced on Cisco Catalyst 2960-X Series Switches and Cisco Catalyst 2960-XR Series Switches.</p>
Flexible NetFlow Lite	Cisco IOS Release 15.0(2)EX1	In Cisco IOS Release 15.0(2)EX1, this feature was introduced on Cisco Catalyst 2960-XR Series Switches.



PART **VIII**

Openflow

- [OpenFlow, on page 709](#)



CHAPTER 39

OpenFlow

- [Finding Feature Information, on page 709](#)
- [Prerequisites for OpenFlow, on page 709](#)
- [Restrictions for OpenFlow, on page 710](#)
- [Information About Open Flow, on page 711](#)
- [Configuring OpenFlow, on page 717](#)
- [Monitoring OpenFlow, on page 721](#)
- [Configuration Examples for OpenFlow, on page 721](#)

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to <http://www.cisco.com/go/cfn>. An account on Cisco.com is not required.

Prerequisites for OpenFlow

The Prerequisites for OpenFlow are as follows:

- A Cisco device and its corresponding operating system that supports the installation of OpenFlow.

Refer to the corresponding release notes for information about which operating system release supports the features and necessary infrastructure.



Note Note: Release notes for [Cisco Catalyst 2960X/XR Series Switches](#)

- A controller installed on a connected server.

Table 74: Controller Support

OpenFlow Version	Supported Controllers
OpenFlow 1.0	Extensible Network Controller (XNC) 1.0, POX, Cisco Open SDN Controller, or Ixia controllers
OpenFlow 1.3	Ixia, CiscoOpen SDN Controller, or OpenDaylight

Restrictions for OpenFlow

The Restrictions for OpenFlow are as listed below:

- OpenFlow supports only a subset of OpenFlow 1.3 functions. For more information, see the Cisco OpenFlow Feature Support section.
- You cannot configure more than one OpenFlow logical switch. The logical switch ID has a value of 1.
- OpenFlow hybrid model (ships-in-the-night) is supported. VLANs configured for OpenFlow logical switch ports should not overlap with regular device interfaces.
- The OpenFlow logical switch ports must not be configured in a mode other than trunk port.
- You cannot configure a bridge domain, Virtual LANs, virtual routing and forwarding (VRF) or port-channel interfaces on an OpenFlow logical switch. You can only configure physical interfaces.
- You cannot make additional configurations to an interface configured as a port of OpenFlow Logical Switch without removing the configuration as a port of OpenFlow Logical Switch.
- In stack scenarios, consisting of active/member switches, whenever the active switch goes down, all current configuration will exist in newly elected active switch. However, the flows have to program again from the controller.
- MIBs and XMLs are not supported.
- Cisco Catalyst 2960X/XR switch supports 1000 L2 flows with EtherType, 200 L2 flows without EtherType, and 500 L3 flows.
- A maximum of 48 ports can be assigned for Openflow operation.
- In general, the maximum sustained flow programming rate from the controller should not exceed 50 (added or deleted) flows per second. For flows that have more than 1 match criteria (more than input port + 1 match), the sustained controller programming rate should not exceed 40 flows per second.
- The maximum burst flow programming rate from the controller should not exceed 1000 flows, spaced by 30-second time intervals. A minimum of 30-second time interval should be maintained between addition or deletion of flows.
- The rate of PACKET_IN messages sent to the controller should be rate-limited to 300 packets per second, using configuration.

Information About Open Flow

Overview of OpenFlow

OpenFlow is a standard communications interface defined between the control and forwarding plane for direct access to and manipulation of the forwarding plane of network devices such as switches and routers from multiple vendors.

OpenFlow Switch Specification Version 1.0.1 (Wire Protocol 0x01), referred to as OpenFlow 1.0, and OpenFlow Switch Specification Version 1.3.0 (Wire Protocol 0x04), referred to as OpenFlow 1.3, are based on the concept of an Ethernet switch with an internal flow table and standardized interface to allow traffic flows on a device to be added or removed. OpenFlow 1.3 defines the communication channel between OpenFlow and controllers.

A generic OpenFlow controller will interact with an specialized OpenFlow agent that translates the OpenFlow configuration into IOS configurations and configures the data plane.

Support of OpenFlow on catalyst 2960X/XR is limited to only software forwarding (due to ASIC limitations). The software forwarding of flows will happen at the OpenFlow agent with support of 12 tuple matches consisting of single table with both L2 and L3 fields together. The match criteria can be match on all 12 tuple fields or any of the 12 tuple fields.

The corresponding actions to the matching criteria can be:

- Push / Pop of Vlan
- Output the packet to port
- Drop the packet
- Set/Decrement IP TTL value
- Modify of L2/L3/L4 fields of Ethernet frame

The Physical ports can be configured as OpenFlow ports or as normal port. The flows in the flow table will be installed based on the priority of the flow.



Note Priority 0 flows are not supported.

Cisco supports a subset of OpenFlow 1.0 and OpenFlow 1.3 functions. A controller can be Extensible Network Controller (XNC) 1.0, or any controller compliant with OpenFlow 1.3.

OpenFlow Controller Operation

OpenFlow controller (referred to as controller) controls the switch and inserts flows with a subset of OpenFlow 1.3 and 1.0 match and action criteria through OpenFlow logical switch.

Cisco OpenFlow Feature Support

The following is a subset of OpenFlow 1.3 and OpenFlow 1.0 functions that are supported by OpenFlow.

Table 75: Cisco OpenFlow Feature Support

Feature	Notes
Configuration of physical interfaces as OpenFlow logical switch ports	<p>Bridge domain, Virtual LANs and Virtual Routing and Forwarding (VRF), and port-channel interfaces are not supported.</p> <p>Only L2 interfaces can be OpenFlow logical switch ports.</p>
Supported OpenFlow message types	<p>Controller to switch:</p> <ul style="list-style-type: none"> • Handshake • Switch Configuration • Modify State (Port Modification message is not supported) • Read State • Packet-Out • Barrier <p>Asynchronous messages:</p> <ul style="list-style-type: none"> • Packet-In • Flow Removed • Port Status • Error <p>Symmetric messages:</p> <ul style="list-style-type: none"> • Hello • Echo Request • Echo Reply • Vendor
Connection to controllers	<p>You can connect up to eight controllers.</p> <p>Connection to the controller through a management interface or a switched virtual interface (SVI) is supported.</p> <p>Connection via TCP and TLS is supported.</p>

Feature	Notes
Multiple actions	<p>If multiple actions are associated with a flow, they are processed in the order specified. The output action should be the last action in the action list. Any action after the output action is not supported, and can cause the flow to fail and return an error to the controller.</p> <p>Flows defined on the controller must follow the these guidelines:</p> <ul style="list-style-type: none"> • The flow can have only one output action. • Some action combinations which are not supported may be rejected at flow programming time. • The flow should not have an output-to-controller action in combination with other rewrite actions.
Supported OpenFlow counters	<p>Per Table—Active entries, packet lookups, and packet matches.</p> <p>Per Flow—Received Packets, Received bytes, Duration (seconds), Duration (milliseconds).</p> <p>Per Port—Received or transmitted packets, and bytes.</p> <p>Per Controller— Flow addition, modification, deletion, error messages, echo requests or replies, barrier requests or replies, connection attempts, successful connections, packet in or packet out.</p>
Default forwarding rule	<p>All packets that cannot be matched to programmed flows are dropped by default. You can configure sending unmatched packets to the controller. You can modify the default action taken on unmatched packets either using the default-miss command or by the controller.</p>
Idle timeout	<p>A minimum Idle timeout of 14 seconds is supported for 700 flows and 48 ports.</p> <p>The statistics collection interval influences the minimum idle timeout. When the interval is set to 7 seconds, the timeout is a minimum of 14 seconds. 700 flows are supported with the 14-second idle timeout.</p> <p>When using an idle timeout of less than 25 seconds, the number of L3 flows should be limited to 700.</p>

Supported Match and Actions and Pipelines

Table 76: Supported Match and Actions and Pipelines

Feature	Notes
Pipelines	Pipelines are mandatory for logical switch. The logical switch supports only pipeline 1. The logical switch supports only table 0.

Feature	Notes
Forwarding Table	

Feature	Notes
	<p>Match Criteria:</p> <ul style="list-style-type: none"> • Input Port • Ethernet type • Source Mac Address • Dest Mac Address • VLAN ID • IP TOS (DSCP bits) • IP Protocol (except for lower 8 bits of ARP code) • IPv4 Source Address • IPv4 Destination Address • Layer 4 Source Port • Layer 4 Destination Port • IPv6 Source Address • IPv6 Destination Address <p>Action Criteria:</p> <ul style="list-style-type: none"> • Forward: Controller • Forward: Port • Forward: Drop • Forward: Controller + Port • Set VLAN ID • New VLAN ID • Replace VLAN ID • Strip VLAN Header • Modify Source MAC • Modify Destination MAC • Modify IPv4 Source Address • Modify IPv4 Destination Address • Modify IPv4 TOS bits • Modify L4 source port • Modify L4 destination port • Decrement TTL

Feature	Notes
Number of flows	1000
Configuration of VLANs	VLAN range is from 1 to 4094.

Configuring OpenFlow

To configure OpenFlow logical switch and the IP address of a controller, perform this task:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **feature openflow**
4. **openflow**
5. **switch** *logical-switch-id* **pipeline** *logical-id*
6. **controller** [**ipv4** *ip-address*] [**port** *tcp-port*] [**vrf** *vrf-name*] [**security** {**none** | **tls**}]
7. **of-port interface** *interface-name*
8. **default-miss** *action-for-unmatched-flows*
9. **protocol-version** {**1.1** | **1.3** | **negotiate**}
10. **shutdown**
11. **datapath-id** *datapath-id*
12. **tls trust-point local** *local-trust-point* **remote** *remote-trust-point*
13. **probe-interval** *probe-interval*
14. **rate-limit packet_in** *controllet-packet-rate* **burst** *maximum-packets-to-controller*
15. **max-backoff** *backoff-timer*
16. **logging flow-mod**
17. **statistics collection-interval** *interval*
18. **end**
19. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.

	Command or Action	Purpose
Step 3	feature openflow Example: Device(config)# feature openflow	Enables Open Flow Agent support on the switch.
Step 4	openflow Example: Device(config)# openflow	Enables Open Flow Agent support on the switch.
Step 5	switch <i>logical-switch-id</i> pipeline <i>logical-id</i> Example: Device(config-ofa-switch)# switch 1 pipeline 1	Specifies an ID for a logical switch that is used for OpenFlow switching and enters logical switch configuration mode. The only logical switch ID supported is 1. Configures a pipeline. This step is mandatory for a logical switch configuration. The only pipeline ID supported is 1.
Step 6	controller [ipv4 <i>ip-address</i>] [port <i>tcp-port</i>] [vrf <i>vrf-name</i>] [security {none tls}] Example: Device(config-ofa-switch)# controller ipv4 10.1.1.1 tcp 6633	Specifies the IPv4 address, port number used by the controller to connect to the logical switch Repeat this step if you need to configure additional controllers. You can configure up to eight controllers. If TLS is used in this step, configure TLS trustpoints in the next step. If unspecified, by default, Controllers use TCP port 6633. A connection to a controller is initiated by the logical switch.
Step 7	of-port interface <i>interface-name</i> Example: Device(config-ofa-switch)# of-port interface GigabitEthernet1/0/23 Device(config-ofa-switch)# of-port interface TenGigabitEthernet1/1/2	Adds interfaces to the logical switch configuration. Observe these guidelines: <ul style="list-style-type: none"> • Do not abbreviate the interface type. Ensure that the interface type is spelled out completely and is as shown in the examples. • If the keyword is abbreviated, the interface is not configured. • The interface must be designated for the OpenFlow logical switch only. Repeat this step to configure additional interfaces.
Step 8	default-miss <i>action-for-unmatched-flows</i> Example:	Configures the action to be taken for packets that do not match any of the flow defined. The supported options are: <ul style="list-style-type: none"> • forward the packets using the normal routing tables

	Command or Action	Purpose
	<pre>Device(config-ofa-switch)# default-miss continue-controller</pre>	<ul style="list-style-type: none"> • forward the packets to the controller • drop the packets <p>The default option is to forward the packets using the normal routing tables.</p>
Step 9	<p>protocol-version {1.1 1.3 negotiate}</p> <p>Example:</p> <pre>Device(config-ofa-switch)# protocol-version negotiate</pre>	<p>Configures the protocol version. Supported values are:</p> <ul style="list-style-type: none"> • 1.0—Configures device to connect to 1.0 controllers only. • 1.3—Configures device to connect to 1.3 controllers only.. • negotiate—Negotiates the protocol version with the controller. Device uses 1.3 for negotiation. <p>The default value is 1.0.</p>
Step 10	<p>shutdown</p> <p>Example:</p> <pre>Device(config-ofa-switch)# shutdown</pre>	<p>Disables a logical switch, bringing down the tcp/ip connection and removing flows from the dataplane.</p>
Step 11	<p>datapath-id <i>datapath-id</i></p> <p>Example:</p> <pre>Device(config-ofa-switch)# datapath-id 0x222</pre>	<p>Configures a unique datapath ID for the switch.</p> <p>This step is mandatory for a logical switch configuration.</p> <p>Enter a 64-bit hexadecimal value.</p>
Step 12	<p>tls trust-point local <i>local-trust-point</i> remote <i>remote-trust-point</i></p> <p>Example:</p> <pre>Device(config-ofa-switch)# tls trust-point local myCA remote myCA</pre>	<p>(Optional) Specifies the local and remote TLS trustpoints to be used for the controller connection.</p>
Step 13	<p>probe-interval <i>probe-interval</i></p> <p>Example:</p> <pre>Device(config-ofa-switch)# probe-interval 7</pre>	<p>(Optional) Configures the interval (in seconds) at which the controller is probed.</p> <p>After the configured interval of time passes, if the switch has not received any messages from the controller, the switch sends an echo request (echo_request) to the controller. It should normally receive an echo reply (echo_reply). If no message is seen for the duration of another probe interval, the switch presumes that the controller is down and disconnects the controller connection. The switch tries to reconnect periodically.</p> <p>The default value is 5 seconds; the range is from 5 to 65535 seconds.</p>

	Command or Action	Purpose
Step 14	<p>rate-limit packet_in <i>controllet-packet-rate</i> burst <i>maximum-packets-to-controller</i></p> <p>Example:</p> <pre>Device(config-ofa-switch)# rate-limit packet_in 300 burst 50</pre>	<p>(Optional) Configures the maximum packet rate sent to the controller and the maximum packets burst sent to the controller in a second.</p> <p>The default value is zero, that is, an indefinite packet rate and packet burst is permitted.</p> <p>This rate limit is for OpenFlow. It is not related to the rate limit of the device (data plane) configured by COPP.</p>
Step 15	<p>max-backoff <i>backoff-timer</i></p> <p>Example:</p> <pre>Device(config-ofa-switch)# max-backoff 8</pre>	<p>(Optional) Configures the duration (in seconds) for which the device must wait before attempting to initiate a connection with the controller.</p> <p>The device initially tries to initiate connection frequently, as the number of unsuccessful attempts increases, the device tries less frequently, that is, the waiting period between attempts also increases. The backoff timer configures the maximum period that the device waits in-between each retry.</p> <p>The default value is 8 seconds; the range is from 1 to 65535 seconds.</p>
Step 16	<p>logging flow-mod</p> <p>Example:</p> <pre>Device(config-ofa-switch)# logging flow-mod</pre>	<p>(Optional) Enables logging of flow changes, including addition, deletion, and modification of flows.</p> <p>Logging of flow changes is a CPU intensive activity and should not be enabled for a large number of flows.</p> <p>Logging of flow changes is disabled by default.</p> <p>Flow changes are logged in syslog and can be viewed using the show logging command.</p>
Step 17	<p>statistics collection-interval <i>interval</i></p> <p>Example:</p> <pre>Device(config-ofa-switch)# statistics collection-interval 7</pre>	<p>Configures the statistics collection interval (in seconds) for all configured flows of OpenFlow. Observe these guidelines:</p> <ul style="list-style-type: none"> • The default interval value is 7 seconds. • The minimum interval is 7 seconds; the maximum is 82 seconds. • You can also specify a value of 0, this disables statistics collection. • Flows with an idle timeout value less than 2 * interval are rejected. <p>Configured interval value is displayed in the output of the show openflow switch 1 command.</p>
Step 18	<p>end</p> <p>Example:</p>	<p>Returns to privileged EXEC mode.</p>

	Command or Action	Purpose
	Device(config-ofa-switch)# end	Alternatively, you can also press Ctrl-Z to exit global configuration mode.
Step 19	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Monitoring OpenFlow

You can monitor OpenFlow parameters using the following commands:

Commands	Description
show openflow switch <i>switch-id</i>	Displays information related to OpenFlow on the logical switch.
show openflow switch <i>switch-id</i> controllers [stats]	Displays information related to the connection status between an OpenFlow logical switch and connected Controllers.
show openflow switch <i>switch-id</i> ports	Displays the mapping between physical device interfaces and ports of OpenFlow logical switch.
show openflow <i>switch-id</i> flows	Displays flows defined for the device by controllers.
show openflow switch <i>switch-id</i> stats	Displays send and receive statistics for each port defined for an OpenFlow logical switch.
show running-config section openflow	Displays configurations made for OpenFlow.
show openflow hardware capabilities	Displays OpenFlow hardware configurations.

Configuration Examples for OpenFlow

This example shows how you can view information related to OpenFlow on the logical switch.

```
Device#show openflow switch 1
```

```
Logical Switch Context
Id: 1
Switch type: Forwarding
Pipeline id: 1
Data plane: secure
Table-Miss default: drop
Configured protocol version: Negotiate
Config state: no-shutdown
```

```

Working state: enabled
Rate limit (packet per second): 0
Burst limit: 0
Max backoff (sec): 8
Probe interval (sec): 5
TLS local trustpoint name: not configured
TLS remote trustpoint name: not configured
Logging flow changes: Disabled
Stats collect interval (sec): 7
Stats collect Max flows: 1000
Stats collect period (sec): 1
Minimum flow idle timeout (sec): 14
OFA Description:
  Manufacturer: Cisco Systems, Inc.
  Hardware: WS-C2960X-48LPS-L
  Software: Cisco IOS Software, C2960X Software (C2960X-UNIVERSALK9-M),
Version 15.2(5.1.50)E, TEST ENGINEERING ESTG_WEEKLY BUILD, synced to
V152_4_1_20_E1| openvswitch 2.1
  Serial Num: FCW1910B5QR
  DP Description: 2960xr:sw1
OF Features:
  DPID: 0x00000000000000251
  Number of tables: 1
  Number of buffers: 256
  Capabilities: FLOW_STATS TABLE_STATS PORT_STATS
Controllers:
  10.106.253.118:6653, Protocol: TCP, VRF: default
Interfaces:
  GigabitEthernet1/0/1
  GigabitEthernet1/0/2

```

This example shows how you can view information related to the connection status between an OpenFlow logical switch and connected Controllers.

```
Device#show openflow switch 1 controllers
```

```

Logical Switch Id: 1
Total Controllers: 1
Controller: 1
  10.106.253.118:6653
  Protocol: tcp
  VRF: default
  Connected: Yes
  Role: Equal
  Negotiated Protocol Version: OpenFlow 1.3
  Last Alive Ping: 2016-04-03 18:40:48 UTC
  state: ACTIVE
  sec_since_connect: 192038

```

```
Device#show openflow switch 1 controllers stats
```

```

Logical Switch Id: 1
Total Controllers: 1

Controller: 1
address : tcp:10.106.253.118:6653
connection attempts : 9
successful connection attempts : 1
flow adds : 1
flow mods : 0
flow deletes : 0
flow removals : 0
flow errors : 0
flow unencodable errors : 0
total errors : 0
echo requests : rx: 0, tx:0
echo reply : rx: 0, tx:0
flow stats : rx: 64004, tx:64004
barrier : rx: 0, tx:0
packet-in/packet-out : rx: 0, tx:0

```

This example shows how you can view the mapping between physical device interfaces and ports of OpenFlow logical switch.

```
Device#show openflow switch 1 ports
```

```

Logical Switch Id: 1
Port Interface Name Config-State Link-State Features
1 Gi1/0/1 PORT_UP LINK_UP 1GB-FD
2 Gi1/0/2 PORT_UP LINK_UP 1GB-FD

```

This example shows how you can view flows defined for the device by controllers.

```
Device#show openflow switch 1 flows
```

```

Logical Switch Id: 1
Total flows: 2

Flow: 1
Match:
  Actions: drop
  Priority: 0
  Table: 0
  Cookie: 0x0
  Duration: 4335.022s
  Number of packets: 18323
  Number of bytes: 1172672

Flow: 2
Match: ipv6
  Actions: output:2
  Priority: 1

```

```

Table: 0
Cookie: 0x0
Duration: 727.757s
Number of packets: 1024
Number of bytes: 131072

```

This example shows how you can view the send and receive statistics for each port defined for an OpenFlow logical switch.

```
Device#show openflow switch 1 stats
```

```

Logical Switch Id: 1
Total ports: 2
  Port 1: rx
           tx
  Port 2: rx
           tx
Total tables: 1
Table 0: Main
Wildcards = 0x00000
Max entries = 1000
Active entries = 2
Number of lookups = 0
Number of matches = 0

```

This example shows how you can view configurations made for OpenFlow.

```
Device#show running-config | section openflow
```

```

feature openflow
mode openflow
mode openflow
openflow
switch 1 pipeline 1
controller ipv4 10.106.253.118 port 6653 security none
of-port interface GigabitEthernet1/0/1
of-port interface GigabitEthernet1/0/2
datapath-id 0x251

```

This example shows how you can view OpenFlow hardware configurations.

```
Device#show openflow hardware capabilities
```

```

Max Flow Batch Size: 100
Statistics Max Polling Rate (flows/sec): 1024
Max Interfaces: 1000
Aggregated Statistics: YES
Pipeline ID: 1
Pipeline Max Flows: 1000

```

```

Pipeline Default Statistics Collect Interval: 7
Flow table ID: 0

Max Flow Batch Size: 100
Max Flows: 1000
Bind Subintfs: FALSE
Primary Table: TRUE
Table Programmable: TRUE
Miss Programmable: TRUE
Number of goto tables: 0
Goto table id:
Stats collection time for full table (sec): 1
Match Capabilities          Match Types
-----
ethernet mac destination    optional
ethernet mac source         optional
ethernet type               optional
VLAN ID                     optional
IP DSCP                     optional
IP protocol                 optional
IPv4 source address         lengthmask
IPv4 destination address    lengthmask
ipv6 source addresss        lengthmask
ipv6 destination address    lengthmask
source port                 optional
destination port            optional
in port (virtual or physical) optional

Actions                      Count Limit Order
-----
set eth source mac           1          10
set eth destination mac      1          10
set vlan id                  1          10
set IPv4 source address      1          10
set IPv4 destination address 1          10
set IP dscp                  1          10
set TCP source port          1          10
set TCP destination port     1          10
set UDP source port          1          10
set UDP destination port     1          10
pop vlan tag                 1          10
set qos group                1          10
drop packet                  1          100
specified interface          1          100
controller                   1          100
divert a copy of pkt to application 1          100

Miss actions                  Count Limit Order
-----

```

```
drop packet          1          100
controller          1          100
```




PART IX

Quality of Service

- [Configuring QoS, on page 729](#)
- [Configuring Auto-QoS, on page 829](#)



CHAPTER 40

Configuring QoS

- [Prerequisites for QoS, on page 729](#)
- [Restrictions for QoS, on page 731](#)
- [Information About QoS, on page 732](#)
- [How to Configure QoS, on page 753](#)
- [Monitoring Standard QoS, on page 816](#)
- [Configuration Examples for QoS, on page 817](#)
- [Where to Go Next, on page 826](#)
- [Additional References, on page 827](#)
- [Feature History and Information for QoS, on page 827](#)

Prerequisites for QoS

Before configuring standard QoS, you must have a thorough understanding of these items:

- The types of applications used and the traffic patterns on your network.
- Traffic characteristics and needs of your network. For example, is the traffic on your network bursty? Do you need to reserve bandwidth for voice and video streams?
- Bandwidth requirements and speed of the network.
- Location of congestion points in the network.

You can configure QoS on physical ports and on switch virtual interfaces (SVIs). Other than to apply policy maps, you configure the QoS settings, such as classification, queueing, and scheduling, the same way on physical ports and SVIs. When configuring QoS on a physical port, you apply a nonhierarchical policy map. When configuring QoS on an SVI, you apply a nonhierarchical or a hierarchical policy map.

QoS ACL Guidelines

Follow these guidelines when configuring QoS with access control lists (ACLs):

- It is not possible to match IP fragments against configured IP extended ACLs to enforce QoS. IP fragments are sent as best-effort. IP fragments are denoted by fields in the IP header.
- Only one ACL per class map and only one **match** class-map configuration command per class map are supported. The ACL can have multiple ACEs, which match fields against the contents of the packet.

- A trust statement in a policy map requires multiple hardware entries per ACL line. If an input service policy map contains a trust statement in an ACL, the access list might be too large to fit into the available QoS hardware memory, and an error can occur when you apply the policy map to a port. Whenever possible, you should minimize the number of lines in a QoS ACL.

Applying QoS on Interfaces Guidelines

These are the guidelines for configuring QoS on physical ports and SVIs (Layer 3 VLAN interfaces):

- You can configure QoS on physical ports and SVIs. When configuring QoS on physical ports, you create and apply nonhierarchical policy maps. When configuring QoS on SVIs, you can create and apply nonhierarchical and hierarchical policy maps.
- Incoming traffic is classified, policed, and marked down (if configured) regardless of whether the traffic is bridged, routed, or sent to the CPU. It is possible for bridged frames to be dropped or to have their DSCP and CoS values modified.
- Follow these guidelines when configuring policy maps on physical ports or SVIs:
 - You cannot apply the same policy map to a physical port and to an SVI.
 - If VLAN-based QoS is configured on a physical port, the switch removes all the port-based policy maps on the port. The traffic on this physical port is now affected by the policy map attached to the SVI to which the physical port belongs.
 - In a hierarchical policy map attached to an SVI, you can only configure an individual policer at the interface level on a physical port to specify the bandwidth limits for the traffic on the port. The ingress port must be configured as a trunk or as a static-access port. You cannot configure policers at the VLAN level of the hierarchical policy map.
 - The switch does not support aggregate policers in hierarchical policy maps.
 - After the hierarchical policy map is attached to an SVI, the interface-level policy map cannot be modified or removed from the hierarchical policy map. A new interface-level policy map also cannot be added to the hierarchical policy map. If you want these changes to occur, the hierarchical policy map must first be removed from the SVI. You also cannot add or remove a class map specified in the hierarchical policy map.

Policing Guidelines

- The port ASIC device, which controls more than one physical port, supports 256 policers (255 user-configurable policers plus 1 policer reserved for system internal use). The maximum number of user-configurable policers supported per port is 63. Policers are allocated on demand by the software and are constrained by the hardware and ASIC boundaries.

For example, you could configure 32 policers on a Gigabit Ethernet port and 7 policers on a 10-Gigabit Ethernet port, or you could configure 64 policers on a Gigabit Ethernet port and 4 policers on a 10-Gigabit Ethernet port. Policers are allocated on demand by the software and are constrained by the hardware and ASIC boundaries.

You cannot reserve policers per port; there is no guarantee that a port will be assigned to any policer.

- Only one policer is applied to a packet on an ingress port. Only the average rate and committed burst parameters are configurable.
- You can create an aggregate policer that is shared by multiple traffic classes within the same nonhierarchical policy map. However, you cannot use the aggregate policer across different policy maps.
- On a port configured for QoS, all traffic received through the port is classified, policed, and marked according to the policy map attached to the port. On a trunk port configured for QoS, traffic in all VLANs received through the port is classified, policed, and marked according to the policy map attached to the port.
- If you have EtherChannel ports configured on your switch, you must configure QoS classification, policing, mapping, and queueing on the individual physical ports that comprise the EtherChannel. You must decide whether the QoS configuration should match on all ports in the EtherChannel.
- If you need to modify a policy map of an existing QoS policy, first remove the policy map from all interfaces, and then modify or copy the policy map. After you finish the modification, apply the modified policy map to the interfaces. If you do not first remove the policy map from all interfaces, high CPU usage can occur, which, in turn, can cause the console to pause for a very long time.

General QoS Guidelines

- Control traffic (such as spanning-tree bridge protocol data units [BPDUs] and routing update packets) received by the switch are subject to all ingress QoS processing.
- You are likely to lose data when you change queue settings; therefore, try to make changes when traffic is at a minimum.
- A switch that is running the IP services feature set supports QoS DSCP and IP precedence matching in policy-based routing (PBR) route maps with these limitations:
 - You cannot apply QoS DSCP mutation maps and PBR route maps to the same interface.
 - You cannot configure DSCP transparency and PBR DSCP route maps on the same switch.

Restrictions for QoS

The following are the restrictions for QoS:

- Ingress queueing and scheduling are not supported on the switch.
- IPv6 QoS is not supported on switches running the LAN base feature set.
- IPv6 ACLs are not supported on switches running the LAN base feature set.
- The switch supports 3 templates: default, vlan, and IPv4. Both the default and vlan templates support IPv6. The IPv4 template does not support IPv6.
- You can configure only individual policers on an SVI.
- For the **class-map** [**match-all** | **match-any**] *class-map-name* global configuration command, because only one **match** command per class map is supported, the **match-all** and **match-any** keywords function the same.

- The number of policers supported is 63 irrespective of the number of SVIs configured. For example, if two SVIs are configured, the number of policers supported is 63 for both SVIs together.

Information About QoS

QoS Implementation

Typically, networks operate on a best-effort delivery basis, which means that all traffic has equal priority and an equal chance of being delivered in a timely manner. When congestion occurs, all traffic has an equal chance of being dropped.

When you configure the QoS feature, you can select specific network traffic, prioritize it according to its relative importance, and use congestion-management and congestion-avoidance techniques to provide preferential treatment. Implementing QoS in your network makes network performance more predictable and bandwidth utilization more effective.

The QoS implementation is based on the Differentiated Services (Diff-Serv) architecture, a standard from the Internet Engineering Task Force (IETF). This architecture specifies that each packet is classified upon entry into the network.

The classification is carried in the IP packet header, using 6 bits from the deprecated IP type of service (ToS) field to carry the classification (*class*) information. Classification can also be carried in the Layer 2 frame.

Figure 70: QoS Classification Layers in Frames and Packets

The special bits in the Layer 2 frame or a Layer 3 packet are shown in the following

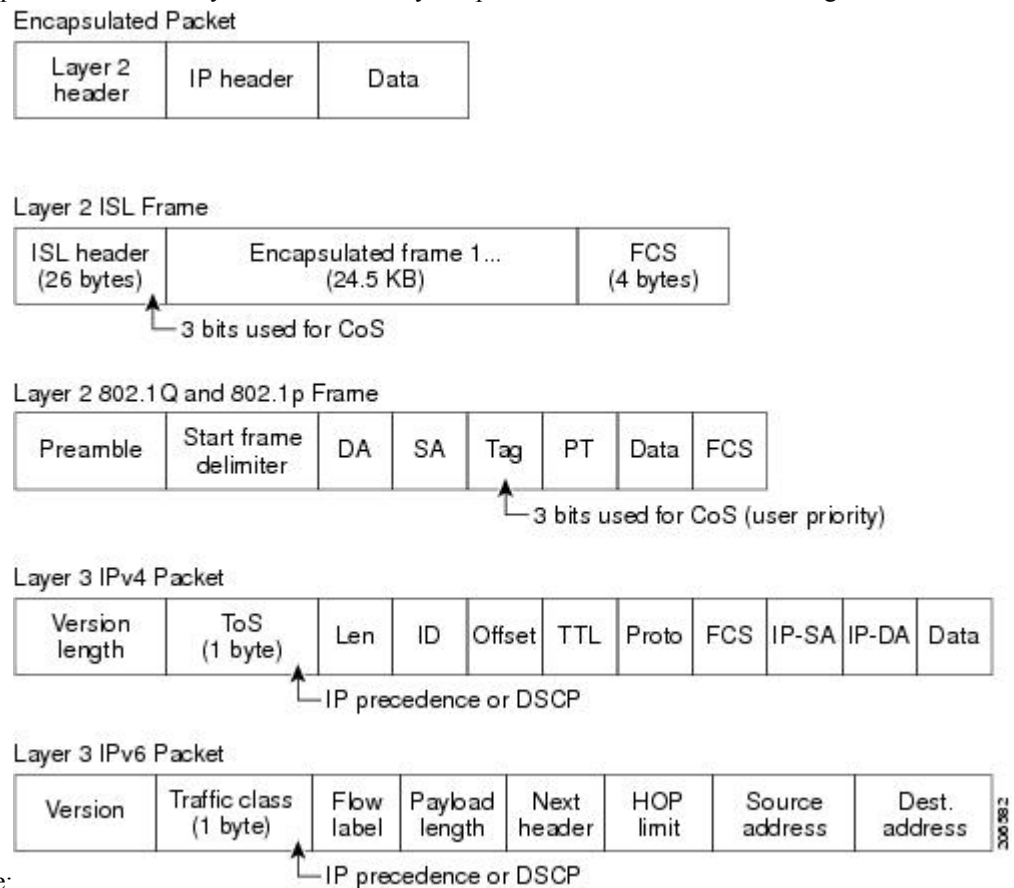


figure:

Layer 2 Frame Prioritization Bits

Layer 2 Inter-Switch Link (ISL) frame headers have a 1-byte User field that carries an IEEE 802.1p class of service (CoS) value in the three least-significant bits. On ports configured as Layer 2 ISL trunks, all traffic is in ISL frames.

Layer 2 802.1Q frame headers have a 2-byte Tag Control Information field that carries the CoS value in the three most-significant bits, which are called the User Priority bits. On ports configured as Layer 2 802.1Q trunks, all traffic is in 802.1Q frames except for traffic in the native VLAN.

Other frame types cannot carry Layer 2 CoS values.

Layer 2 CoS values range from 0 for low priority to 7 for high priority.

Layer 3 Packet Prioritization Bits

Layer 3 IP packets can carry either an IP precedence value or a Differentiated Services Code Point (DSCP) value. QoS supports the use of either value because DSCP values are backward-compatible with IP precedence values.

IP precedence values range from 0 to 7. DSCP values range from 0 to 63.

End-to-End QoS Solution Using Classification

All switches and routers that access the Internet rely on the class information to provide the same forwarding treatment to packets with the same class information and different treatment to packets with different class information. The class information in the packet can be assigned by end hosts or by switches or routers along the way, based on a configured policy, detailed examination of the packet, or both. Detailed examination of the packet is expected to occur closer to the edge of the network, so that the core switches and routers are not overloaded with this task.

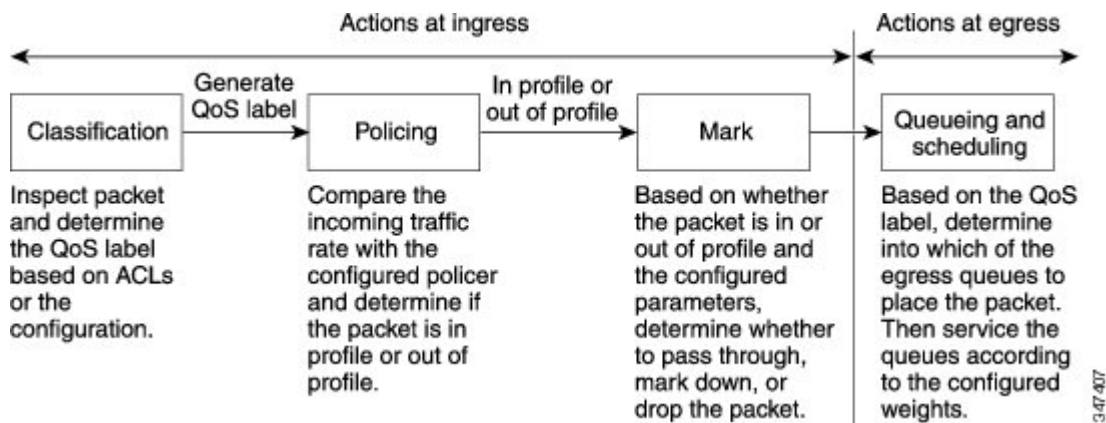
Switches and routers along the path can use the class information to limit the amount of resources allocated per traffic class. The behavior of an individual device when handling traffic in the Diff-Serv architecture is called per-hop behavior. If all devices along a path provide a consistent per-hop behavior, you can construct an end-to-end QoS solution.

Implementing QoS in your network can be a simple task or complex task and depends on the QoS features offered by your internetworking devices, the traffic types and patterns in your network, and the granularity of control that you need over incoming and outgoing traffic.

QoS Basic Model

To implement QoS, the switch must distinguish packets or flows from one another (classify), assign a label to indicate the given quality of service as the packets move through the switch, make the packets comply with the configured resource usage limits (police and mark), and provide different treatment (queue and schedule) in all situations where resource contention exists. The switch also needs to ensure that traffic sent from it meets a specific traffic profile (shape).

Figure 71: QoS Basic Model



Actions at Ingress Port

Actions at the ingress port include classifying traffic, policing, marking, and scheduling:

- Classifying a distinct path for a packet by associating it with a QoS label. The switch maps the CoS or DSCP in the packet to a QoS label to distinguish one kind of traffic from another. The QoS label that is generated identifies all future QoS actions to be performed on this packet.
- Policing determines whether a packet is in or out of profile by comparing the rate of the incoming traffic to the configured policer. The policer limits the bandwidth consumed by a flow of traffic. The result is passed to the marker.

- Marking evaluates the policer and configuration information for the action to be taken when a packet is out of profile and determines what to do with the packet (pass through a packet without modification, marking down the QoS label in the packet, or dropping the packet).



Note Queueing and scheduling are only supported at egress and not at ingress on the switch.

Actions at Egress Port

Actions at the egress port include queueing and scheduling:

- Queueing evaluates the QoS packet label and the corresponding DSCP or CoS value before selecting which of the four egress queues to use. Because congestion can occur when multiple ingress ports simultaneously send data to an egress port, WTD differentiates traffic classes and subjects the packets to different thresholds based on the QoS label. If the threshold is exceeded, the packet is dropped.
- Scheduling services the four egress queues based on their configured SRR shared or shaped weights. One of the queues (queue 1) can be the expedited queue, which is serviced until empty before the other queues are serviced.

Classification Overview

Classification is the process of distinguishing one kind of traffic from another by examining the fields in the packet. Classification is enabled only if QoS is globally enabled on the switch. By default, QoS is globally disabled, so no classification occurs.

During classification, the switch performs a lookup and assigns a QoS label to the packet. The QoS label identifies all QoS actions to be performed on the packet and from which queue the packet is sent.

The QoS label is based on the DSCP or the CoS value in the packet and decides the queuing and scheduling actions to perform on the packet. The label is mapped according to the trust setting and the packet type as shown in the Classification Flowchart.

You specify which fields in the frame or packet that you want to use to classify incoming traffic.

Non-IP Traffic Classification

The following table describes the non-IP traffic classification options for your QoS configuration.

Table 77: Non-IP Traffic Classifications

Non-IP Traffic Classification	Description
Trust the CoS value	Trust the CoS value in the incoming frame (configure the port to trust CoS), and then use the configurable CoS-to-DSCP map to generate a DSCP value for the packet. Layer 2 ISL frame headers carry the CoS value in the 3 least-significant bits of the 1-byte User field. Layer 2 802.1Q frame headers carry the CoS value in the 3 most-significant bits of the Tag Control Information field. CoS values range from 0 for low priority to 7 for high priority.

Non-IP Traffic Classification	Description
Trust the DSCP or trust IP precedence value	Trust the DSCP or trust IP precedence value in the incoming frame. These configurations are meaningless for non-IP traffic. If you configure a port with either of these options and non-IP traffic is received, the switch assigns a CoS value and generates an internal DSCP value from the CoS-to-DSCP map. The switch uses the internal DSCP value to generate a CoS value representing the priority of the traffic.
Perform classification based on configured Layer 2 MAC ACL	Perform the classification based on a configured Layer 2 MAC access control list (ACL), which can examine the MAC source address, the MAC destination address, and other fields. If no ACL is configured, the packet is assigned 0 as the DSCP and CoS values, which means best-effort traffic. Otherwise, the policy-map action specifies a DSCP or CoS value to assign to the incoming frame.

After classification, the packet is sent to the policing and marking stages.

IP Traffic Classification

The following table describes the IP traffic classification options for your QoS configuration.

Table 78: IP Traffic Classifications

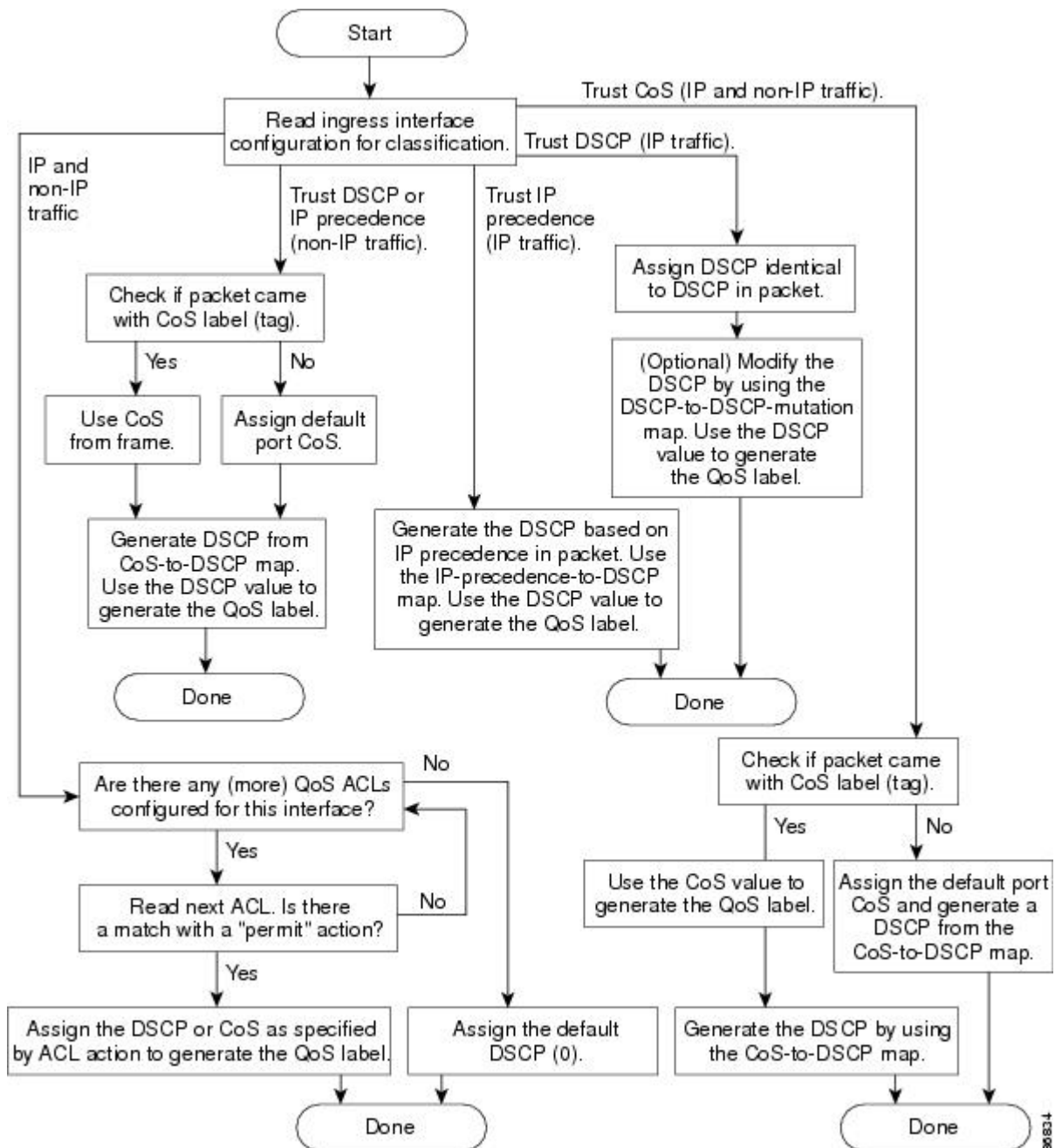
IP Traffic Classification	Description
Trust the DSCP value	Trust the DSCP value in the incoming packet (configure the port to trust DSCP), and assign the same DSCP value to the packet. The IETF defines the 6 most-significant bits of the 1-byte ToS field as the DSCP. The priority represented by a particular DSCP value is configurable. DSCP values range from 0 to 63. You can also classify IP traffic based on IPv6 DSCP. For ports that are on the boundary between two QoS administrative domains, you can modify the DSCP to another value by using the configurable DSCP-to-DSCP-mutation map.
Trust the IP precedence value	Trust the IP precedence value in the incoming packet (configure the port to trust IP precedence), and generate a DSCP value for the packet by using the configurable IP-precedence-to-DSCP map. The IP Version 4 specification defines the 3 most-significant bits of the 1-byte ToS field as the IP precedence. IP precedence values range from 0 for low priority to 7 for high priority. You can also classify IP traffic based on IPv6 precedence.
Trust the CoS value	Trust the CoS value (if present) in the incoming packet, and generate a DSCP value for the packet by using the CoS-to-DSCP map. If the CoS value is not present, use the default port CoS value.

IP Traffic Classification	Description
IP standard or an extended ACL	Perform the classification based on a configured IP standard or an extended ACL, which examines various fields in the IP header. If no ACL is configured, the packet is assigned 0 as the DSCP and CoS values, which means best-effort traffic. Otherwise, the policy-map action specifies a DSCP or CoS value to assign to the incoming frame.
Override configured CoS	Override the configured CoS of incoming packets, and apply the default port CoS value to them. For IPv6 packets, the DSCP value is rewritten by using the CoS-to-DSCP map and by using the default CoS of the port. You can do this for both IPv4 and IPv6 traffic.

After classification, the packet is sent to the policing and marking stages.

Classification Flowchart

Figure 72: Classification Flowchart



Access Control Lists

You can use IP standard, IP extended, or Layer 2 MAC ACLs to define a group of packets with the same characteristics (class). You can also classify IP traffic based on IPv6 ACLs.

In the QoS context, the permit and deny actions in the access control entries (ACEs) have different meanings from security ACLs:

- If a match with a permit action is encountered (first-match principle), the specified QoS-related action is taken.
- If a match with a deny action is encountered, the ACL being processed is skipped, and the next ACL is processed.



Note Deny action is supported in Cisco IOS Release 3.7.4E and later releases.

- If no match with a permit action is encountered and all the ACEs have been examined, no QoS processing occurs on the packet, and the offers best-effort service to the packet.
- If multiple ACLs are configured on a port, the lookup stops after the packet matches the first ACL with a permit action, and QoS processing begins.



Note When creating an access list, note that by default the end of the access list contains an implicit deny statement for everything if it did not find a match before reaching the end.

After a traffic class has been defined with the ACL, you can attach a policy to it. A policy might contain multiple classes with actions specified for each one of them. A policy might include commands to classify the class as a particular aggregate (for example, assign a DSCP) or rate-limit the class. This policy is then attached to a particular port on which it becomes effective.

You implement IP ACLs to classify IP traffic by using the **access-list** global configuration command; you implement Layer 2 MAC ACLs to classify non-IP traffic by using the **mac access-list extended** global configuration command.

Classification Based on Class Maps and Policy Maps

A class map is a mechanism that you use to name a specific traffic flow (or class) and to isolate it from all other traffic. The class map defines the criteria used to match against a specific traffic flow to further classify it. The criteria can include matching the access group defined by the ACL or matching a specific list of DSCP or IP precedence values. If you have more than one type of traffic that you want to classify, you can create another class map and use a different name. After a packet is matched against the class-map criteria, you further classify it through the use of a policy map.

A policy map specifies which traffic class to act on. Actions can include trusting the CoS, DSCP, or IP precedence values in the traffic class; setting a specific DSCP or IP precedence value in the traffic class; or specifying the traffic bandwidth limitations and the action to take when the traffic is out of profile. Before a policy map can be effective, you must attach it to a port.

You create a class map by using the **class-map** global configuration command or the **class** policy-map configuration command. You should use the **class-map** command when the map is shared among many ports. When you enter the **class-map** command, the switch enters the class-map configuration mode. In this mode, you define the match criterion for the traffic by using the **match** class-map configuration command.

You can configure a default class by using the **class class-default** policy-map configuration command. Unclassified traffic (traffic specified in the other traffic classes configured on the policy-map) is treated as default traffic.

You create and name a policy map by using the **policy-map** global configuration command. When you enter this command, the switch enters the policy-map configuration mode. In this mode, you specify the actions to take on a specific traffic class by using the **class**, **trust**, or **set** policy-map configuration and policy-map class configuration commands.

The policy map can contain the **police** and **police aggregate** policy-map class configuration commands, which define the policer, the bandwidth limitations of the traffic, and the action to take if the limits are exceeded.

To enable the policy map, you attach it to a port by using the **service-policy** interface configuration command.

You can apply a nonhierarchical policy map to a physical port or an SVI. However, a hierarchical policy map can only be applied to an SVI. A hierarchical policy map contains two levels. The first level, the VLAN level, specifies the actions to be taken against a traffic flow on the SVI. The second level, the interface level, specifies the actions to be taken against the traffic on the physical ports that belong to the SVI. The interface-level actions are specified in the interface-level policy map.

Policing and Marking Overview

After a packet is classified and has a DSCP-based or CoS-based QoS label assigned to it, the policing and marking process can begin.

Policing involves creating a policer that specifies the bandwidth limits for the traffic. Packets that exceed the limits are *out of profile* or *nonconforming*. Each policer decides on a packet-by-packet basis whether the packet is in or out of profile and specifies the actions on the packet. These actions, carried out by the marker, include passing through the packet without modification, dropping the packet, or modifying (marking down) the assigned DSCP of the packet and allowing the packet to pass through. The configurable policed-DSCP map provides the packet with a new DSCP-based QoS label. Marked-down packets use the same queues as the original QoS label to prevent packets in a flow from getting out of order.



Note All traffic, regardless of whether it is bridged or routed, is subjected to a policer, if one is configured. As a result, bridged packets might be dropped or might have their DSCP or CoS fields modified when they are policed and marked.

You can configure policing (either individual or aggregate policers) on a physical port or an SVI. When configuring policy maps on an SVI, you can create a hierarchical policy map and can define an individual policer only in the secondary interface-level policy map.

After you configure the policy map and policing actions, attach the policy to an ingress port or SVI by using the **service-policy** interface configuration command.

Physical Port Policing

In policy maps on physical ports, you can create the following types of policers:

- **Individual**—QoS applies the bandwidth limits specified in the policer separately to each matched traffic class. You configure this type of policer within a policy map by using the **police** policy-map class configuration command.
- **Aggregate**—QoS applies the bandwidth limits specified in an aggregate policer cumulatively to all matched traffic flows. You configure this type of policer by specifying the aggregate policer name within a policy map by using the **police aggregate** policy-map class configuration command. You specify the bandwidth limits of the policer by using the **mls qos aggregate-policer** global configuration command. In this way, the aggregate policer is shared by multiple classes of traffic within a policy map.



Note You can configure only individual policers on an SVI.

Policing uses a token-bucket algorithm. As each frame is received by the switch, a token is added to the bucket. The bucket has a hole in it and leaks at a rate that you specify as the average traffic rate in bits per second. Each time a token is added to the bucket, the switch verifies that there is enough room in the bucket. If there is not enough room, the packet is marked as nonconforming, and the specified policer action is taken (dropped or marked down).

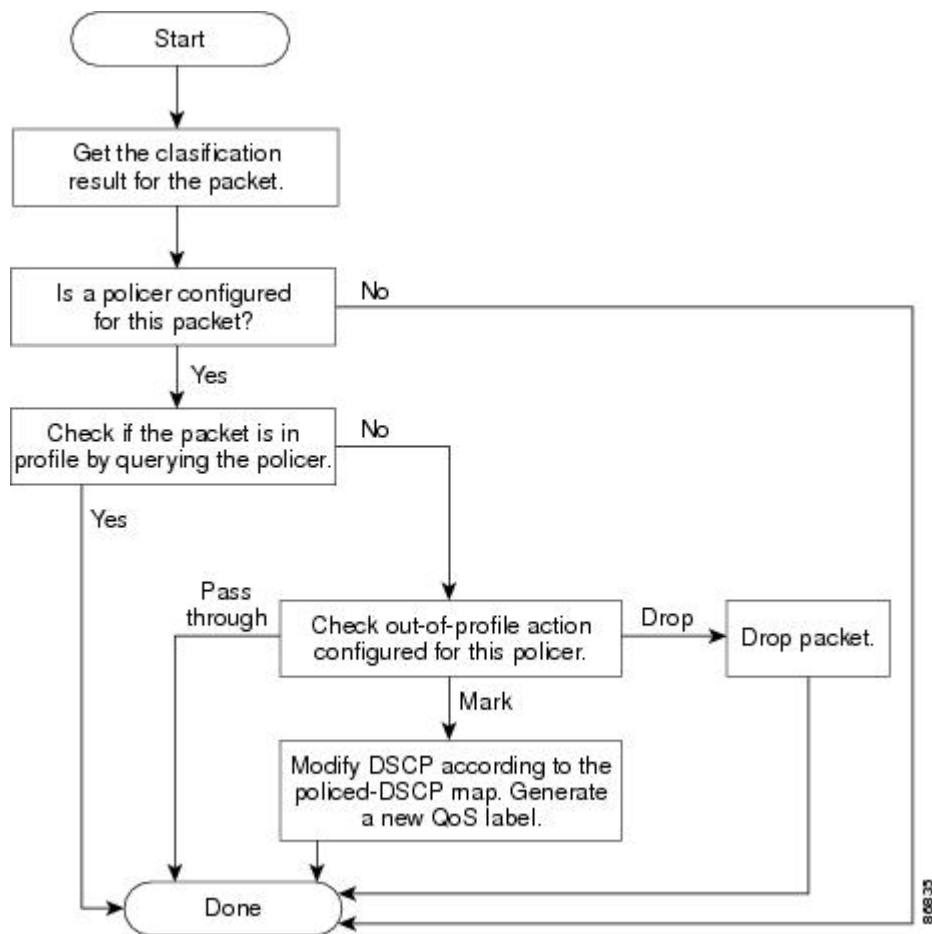
How quickly the bucket fills is a function of the bucket depth (burst-byte), the rate at which the tokens are removed (rate-bps), and the duration of the burst above the average rate. The size of the bucket imposes an upper limit on the burst length and limits the number of frames that can be transmitted back-to-back. If the burst is short, the bucket does not overflow, and no action is taken against the traffic flow. However, if a burst is long and at a higher rate, the bucket overflows, and the policing actions are taken against the frames in that burst.

You configure the bucket depth (the maximum burst that is tolerated before the bucket overflows) by using the `burst-byte` option of the **police** policy-map class configuration command or the **mls qos aggregate-policer** global configuration command. You configure how fast (the average rate) that the tokens are removed from the bucket by using the `rate-bps` option of the **police** policy-map class configuration command or the **mls qos aggregate-policer** global configuration command.

Figure 73: Policing and Marking Flowchart on Physical Ports

The following figure shows the policing and marking process when these types of policy maps are configured:

- A nonhierarchical policy map on a physical port.
- The interface level of a hierarchical policy map attached to an SVI. The physical ports are specified in this secondary policy map.



SVI Policing



Note Before configuring a hierarchical policy map with individual policers on an SVI, you must enable VLAN-based QoS on the physical ports that belong to the SVI. Though a policy map is attached to the SVI, the individual policers only affect traffic on the physical ports specified in the secondary interface level of the hierarchical policy map.

A hierarchical policy map has two levels. The first level, the VLAN level, specifies the actions to be taken against a traffic flow on an SVI. The second level, the interface level, specifies the actions to be taken against the traffic on the physical ports that belong to the SVI and are specified in the interface-level policy map.

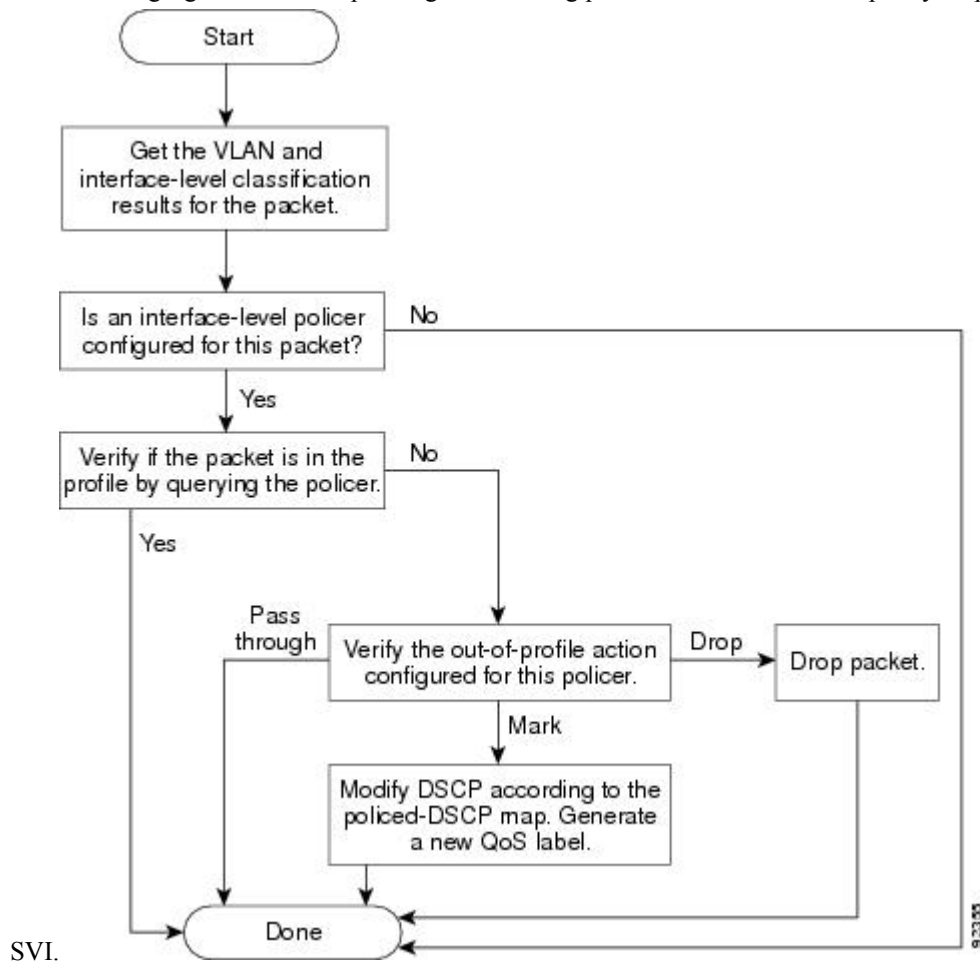
When configuring policing on an SVI, you can create and configure a hierarchical policy map with these two levels:

- **VLAN level**—Create this primary level by configuring class maps and classes that specify the port trust state or set a new DSCP or IP precedence value in the packet. The VLAN-level policy map applies only to the VLAN in an SVI and does not support policers.
- **Interface level**—Create this secondary level by configuring class maps and classes that specify the individual policers on physical ports the belong to the SVI. The interface-level policy map only supports

individual policers and does not support aggregate policers. You can configure different interface-level policy maps for each class defined in the VLAN-level policy map.

Figure 74: Policing and Marking Flowchart on SVIs

The following figure shows the policing and marking process when hierarchical policy maps on an



Mapping Tables Overview

During QoS processing, the switch represents the priority of all traffic (including non-IP traffic) with a QoS label based on the DSCP or CoS value from the classification stage.

The following table describes QoS processing and mapping tables.

Table 79: QoS Processing and Mapping Tables

QoS Processing Stage	Mapping Table Usage
Classification	<p>During the classification stage, QoS uses configurable mapping tables to derive a corresponding DSCP or CoS value from a received CoS, DSCP, or IP precedence value. These maps include the CoS-to-DSCP map and the IP-precedence-to-DSCP map.</p> <p>You configure these maps by using the mls qos map cos-dscp and the mls qos map ip-prec-dscp global configuration commands.</p> <p>On an ingress port configured in the DSCP-trusted state, if the DSCP values are different between the QoS domains, you can apply the configurable DSCP-to-DSCP-mutation map to the port that is on the boundary between the two QoS domains.</p> <p>You configure this map by using the mls qos map dscp-mutation global configuration command.</p>
Policing	<p>During policing stage, QoS can assign another DSCP value to an IP or a non-IP packet (if the packet is out of profile and the policer specifies a marked-down value). This configurable map is called the policed-DSCP map.</p> <p>You configure this map by using the mls qos map policed-dscp global configuration command.</p>
Pre-scheduling	<p>Before the traffic reaches the scheduling stage, QoS stores the packet in an egress queue according to the QoS label. The QoS label is based on the DSCP or the CoS value in the packet and selects the queue through the DSCP output queue threshold maps or through the CoS output queue threshold maps. In addition to an egress queue, the QoS label also identifies the WTD threshold value.</p> <p>You configure these maps by using the mls qos srr-queue { output } dscp-map and the mls qos srr-queue { output } cos-map global configuration commands.</p>

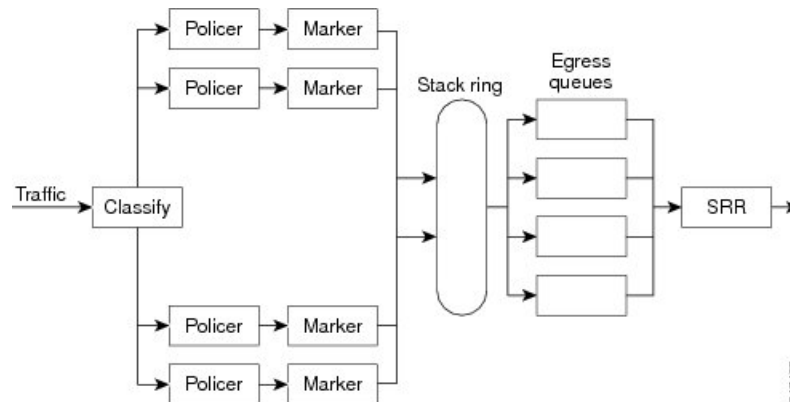
The CoS-to-DSCP, DSCP-to-CoS, and the IP-precedence-to-DSCP maps have default values that might or might not be appropriate for your network.

The default DSCP-to-DSCP-mutation map and the default policed-DSCP map are null maps; they map an incoming DSCP value to the same DSCP value. The DSCP-to-DSCP-mutation map is the only map you apply to a specific port. All other maps apply to the entire switch.

Queueing and Scheduling Overview

The switch has queues at specific points to help prevent congestion.

Figure 75: Egress Queue Location on Switch



Note The switch supports 4 egress queues by default and there is an option to enable a total of 8 egress queues. The 8 egress queue configuration is only supported on a standalone switch.

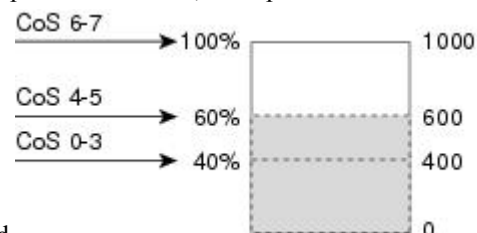
Weighted Tail Drop

As a frame is enqueued to a particular queue, WTD uses the frame's assigned QoS label to subject it to different thresholds. If the threshold is exceeded for that QoS label (the space available in the destination queue is less than the size of the frame), the switch drops the frame.

Each queue has three threshold values. The QoS label determines which of the three threshold values is subjected to the frame. Of the three thresholds, two are configurable (explicit) and one is not (implicit).

Figure 76: WTD and Queue Operation

The following figure shows an example of WTD operating on a queue whose size is 1000 frames. Three drop percentages are configured: 40 percent (400 frames), 60 percent (600 frames), and 100 percent (1000 frames). These percentages indicate that up to 400 frames can be queued at the 40-percent threshold, up to 600 frames at the 60-percent threshold, and up to 1000 frames at the 100-percent



threshold.

In the example, CoS values 6 and 7 have a greater importance than the other CoS values, and they are assigned to the 100-percent drop threshold (queue-full state). CoS values 4 and 5 are assigned to the 60-percent threshold, and CoS values 0 to 3 are assigned to the 40-percent threshold.

Suppose the queue is already filled with 600 frames, and a new frame arrives. It contains CoS values 4 and 5 and is subjected to the 60-percent threshold. If this frame is added to the queue, the threshold will be exceeded, so the switch drops it.

SRR Shaping and Sharing

You can configure SRR on egress queues for sharing or for shaping.

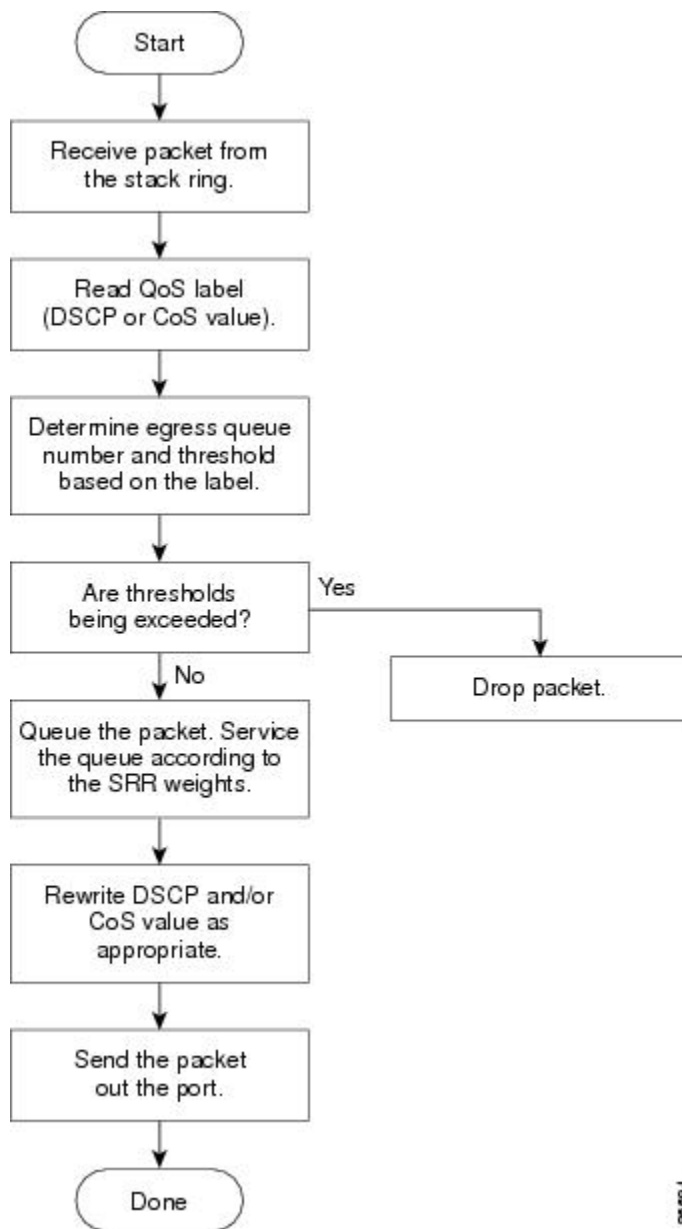
In shaped mode, the egress queues are guaranteed a percentage of the bandwidth, and they are rate-limited to that amount. Shaped traffic does not use more than the allocated bandwidth even if the link is idle. Shaping provides a more even flow of traffic over time and reduces the peaks and valleys of bursty traffic. With shaping, the absolute value of each weight is used to compute the bandwidth available for the queues.

In shared mode, the queues share the bandwidth among them according to the configured weights. The bandwidth is guaranteed at this level but not limited to it. For example, if a queue is empty and no longer requires a share of the link, the remaining queues can expand into the unused bandwidth and share it among them. With sharing, the ratio of the weights controls the frequency of dequeuing; the absolute values are meaningless. Shaping and sharing is configured per interface. Each interface can be uniquely configured.

Queueing and Scheduling on Egress Queues

The following figure shows queueing and scheduling flowcharts for egress ports on the switch.

Figure 77: Queueing and Scheduling Flowchart for Egress Ports on the Switch



Note If the expedite queue is enabled, SRR services it until it is empty before servicing the other three queues.

Egress Expedite Queue

Each port supports four egress queues, one of which (queue 1) can be the egress expedite queue. These queues are assigned to a queue-set. All traffic exiting the switch flows through one of these four queues and is subjected to a threshold based on the QoS label assigned to the packet.



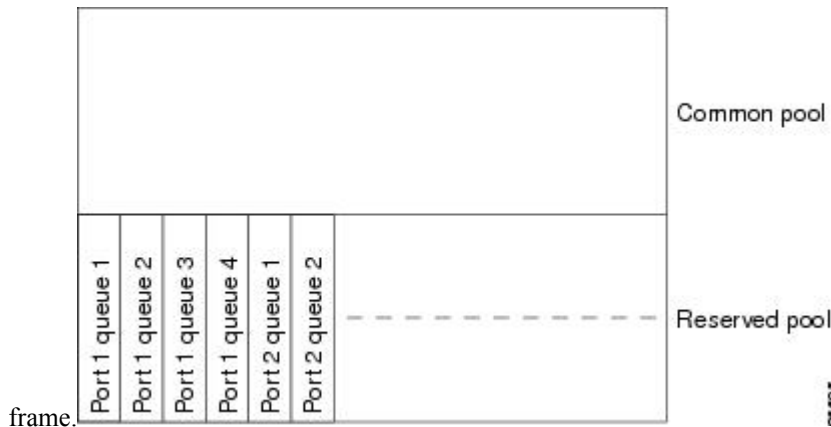
Note If the expedite queue is enabled, SRR services it until it is empty before servicing the other three queues.

Egress Queue Buffer Allocation

The following figure shows the egress queue buffer.

Figure 78: Egress Queue Buffer Allocation

The buffer space is divided between the common pool and the reserved pool. The switch uses a buffer allocation scheme to reserve a minimum amount of buffers for each egress queue, to prevent any queue or port from consuming all the buffers and depriving other queues, and to control whether to grant buffer space to a requesting queue. The switch detects whether the target queue has not consumed more buffers than its reserved amount (under-limit), whether it has consumed all of its maximum buffers (over limit), and whether the common pool is empty (no free buffers) or not empty (free buffers). If the queue is not over-limit, the switch can allocate buffer space from the reserved pool or from the common pool (if it is not empty). If there are no free buffers in the common pool or if the queue is over-limit, the switch drops the



Buffer and Memory Allocation

You guarantee the availability of buffers, set drop thresholds, and configure the maximum memory allocation for a queue-set by using the **mls qos queue-set output *qset-id* threshold *queue-id* drop-threshold1 drop-threshold2 reserved-threshold maximum-threshold** global configuration command. Each threshold value is a percentage of the queue's allocated memory, which you specify by using the **mls qos queue-set output *qset-id* buffers allocation1 ... allocation4** global configuration command. The sum of all the allocated buffers represents the reserved pool, and the remaining buffers are part of the common pool.

Through buffer allocation, you can ensure that high-priority traffic is buffered. For example, if the buffer space is 400, you can allocate 70 percent of it to queue 1 and 10 percent to queues 2 through 4. Queue 1 then has 280 buffers allocated to it, and queues 2 through 4 each have 40 buffers allocated to them.

You can guarantee that the allocated buffers are reserved for a specific queue in a queue-set. For example, if there are 100 buffers for a queue, you can reserve 50 percent (50 buffers). The switch returns the remaining 50 buffers to the common pool. You also can enable a queue in the full condition to obtain more buffers than are reserved for it by setting a maximum threshold. The switch can allocate the needed buffers from the common pool if the common pool is not empty.

Queues and WTD Thresholds

You can assign each packet that flows through the switch to a queue and to a threshold.

Specifically, you map DSCP or CoS values to an egress queue and map DSCP or CoS values to a threshold ID. You use the **mls qos srr-queue output dscp-map queue** *queue-id* {*dscp1...dscp8* | **threshold** *threshold-id* *dscp1...dscp8*} or the **mls qos srr-queue output cos-map queue** *queue-id* {*cos1...cos8* | **threshold** *threshold-id* *cos1...cos8*} global configuration command. You can display the DSCP output queue threshold map and the CoS output queue threshold map by using the **show mls qos maps** privileged EXEC command.

The queues use WTD to support distinct drop percentages for different traffic classes. Each queue has three drop thresholds: two configurable (*explicit*) WTD thresholds and one nonconfigurable (*implicit*) threshold preset to the queue-full state. You assign the two WTD threshold percentages for threshold ID 1 and ID 2. The drop threshold for threshold ID 3 is preset to the queue-full state, and you cannot modify it. You map a port to queue-set by using the **queue-set qset-id** interface configuration command. Modify the queue-set configuration to change the WTD threshold percentages.

Shaped or Shared Mode

SRR services each queue-set in shared or shaped mode. You map a port to a queue-set by using the **queue-set** *qset-id* interface configuration command.

You assign shared or shaped weights to the port by using the **srr-queue bandwidth share** *weight1 weight2 weight3 weight4* or the **srr-queue bandwidth shape** *weight1 weight2 weight3 weight4* interface configuration command.

The buffer allocation together with the SRR weight ratios control how much data can be buffered and sent before packets are dropped. The weight ratio is the ratio of the frequency in which the SRR scheduler sends packets from each queue.

All four queues participate in the SRR unless the expedite queue is enabled, in which case the first bandwidth weight is ignored and is not used in the ratio calculation. The expedite queue is a priority queue, and it is serviced until empty before the other queues are serviced. You enable the expedite queue by using the **priority-queue out** interface configuration command.

You can combine the commands described in this section to prioritize traffic by placing packets with particular DSCPs or CoSs into certain queues, by allocating a large queue size or by servicing the queue more frequently, and by adjusting queue thresholds so that packets with lower priorities are dropped.



Note The egress queue default settings are suitable for most situations. You should change them only when you have a thorough understanding of the egress queues and if these settings do not meet your QoS solution.

Packet Modification

A packet is classified, policed, and queued to provide QoS. The following packet modifications can occur during the process to provide QoS:

- For IP and non-IP packets, classification involves assigning a QoS label to a packet based on the DSCP or CoS of the received packet. However, the packet is not modified at this stage; only an indication of the assigned DSCP or CoS value is carried along.
- During policing, IP and non-IP packets can have another DSCP assigned to them (if they are out of profile and the policer specifies a markdown DSCP). Once again, the DSCP in the packet is not modified, but an indication of the marked-down value is carried along. For IP packets, the packet modification occurs

at a later stage; for non-IP packets the DSCP is converted to CoS and used for queueing and scheduling decisions.

- Depending on the QoS label assigned to a frame and the mutation chosen, the DSCP and CoS values of the frame are rewritten. If you do not configure a table map and if you configure the port to trust the DSCP of the incoming frame, the DSCP value in the frame is not changed, but the CoS is rewritten according to the DSCP-to-CoS map. If you configure the port to trust the CoS of the incoming frame and it is an IP packet, the CoS value in the frame is not changed, but the DSCP might be changed according to the CoS-to-DSCP map.

The input mutation causes the DSCP to be rewritten depending on the new value of DSCP chosen. The set action in a policy map also causes the DSCP to be rewritten.

Standard QoS Default Configuration

Standard QoS is disabled by default.

When QoS is disabled, there is no concept of trusted or untrusted ports because the packets are not modified. The CoS, DSCP, and IP precedence values in the packet are not changed.

Traffic is switched in pass-through mode. The packets are switched without any rewrites and classified as best effort without any policing.

When QoS is enabled using the **mls qos** global configuration command and all other QoS settings are at their defaults, traffic is classified as best effort (the DSCP and CoS value is set to 0) without any policing. No policy maps are configured. The default port trust state on all ports is untrusted.



Note Starting Cisco IOS Release 15.2(1)E, IPv6 QoS is supported on switches running the LAN base license with lanbase-routing template.

Default Egress Queue Configuration

The following tables describe the default egress queue configurations.

The following table shows the default egress queue configuration for each queue-set when QoS is enabled. All ports are mapped to queue-set 1. The port bandwidth limit is set to 100 percent and rate unlimited. Note that for the SRR shaped weights (absolute) feature, a shaped weight of zero indicates that the queue is operating in shared mode. Note that for the SRR shared weights feature, one quarter of the bandwidth is allocated to each queue.

Table 80: Default Egress Queue Configuration

Feature	Queue 1	Queue 2	Queue 3	Queue 4
Buffer allocation	25 percent	25 percent	25 percent	25 percent
WTD drop threshold 1	100 percent	200 percent	100 percent	100 percent
WTD drop threshold 2	100 percent	200 percent	100 percent	100 percent

Feature	Queue 1	Queue 2	Queue 3	Queue 4
Reserved threshold	50 percent	50 percent	50 percent	50 percent
Maximum threshold	400 percent	400 percent	400 percent	400 percent
SRR shaped weights (absolute)	25	0	0	0
SRR shared weights	25	25	25	25

The following table shows the default CoS output queue threshold map when QoS is enabled.

Table 81: Default CoS Output Queue Threshold Map

CoS Value	Queue ID–Threshold ID
0, 1	2–1
2, 3	3–1
4	4–1
5	1–1
6, 7	4–1

The following table shows the default DSCP output queue threshold map when QoS is enabled.

Table 82: Default DSCP Output Queue Threshold Map

DSCP Value	Queue ID–Threshold ID
0–15	2–1
16–31	3–1
32–39	4–1
40–47	1–1
48–63	4–1

Default Mapping Table Configuration

The default DSCP-to-DSCP-mutation map is a null map, which maps an incoming DSCP value to the same DSCP value.

The default policed-DSCP map is a null map, which maps an incoming DSCP value to the same DSCP value (no markdown).

DSCP Maps

Default CoS-to-DSCP Map

When DSCP transparency mode is disabled, the DSCP values are derived from CoS as per the following table. If these values are not appropriate for your network, you need to modify them.

Note The DSCP transparency mode is disabled by default. If it is enabled (**no mls qos rewrite ip dscp** interface configuration command), DSCP rewrite will not happen.

Table 83: Default CoS-to-DSCP Map

CoS Value	DSCP Value
0	0
1	8
2	16
3	24
4	32
5	40
6	48
7	56

Default IP-Precedence-to-DSCP Map

You use the IP-precedence-to-DSCP map to map IP precedence values in incoming packets to a DSCP value that QoS uses internally to represent the priority of the traffic. The following table shows the default IP-precedence-to-DSCP map. If these values are not appropriate for your network, you need to modify them.

Table 84: Default IP-Precedence-to-DSCP Map

IP Precedence Value	DSCP Value
0	0
1	8
2	16
3	24
4	32
5	40
6	48

IP Precedence Value	DSCP Value
7	56

Default DSCP-to-CoS Map

You use the DSCP-to-CoS map to generate a CoS value, which is used to select one of the four egress queues. The following table shows the default DSCP-to-CoS map. If these values are not appropriate for your network, you need to modify them.

Table 85: Default DSCP-to-CoS Map

DSCP Value	CoS Value
0–7	0
8–15	1
16–23	2
24–31	3
32–39	4
40–47	5
48–55	6
56–63	7

How to Configure QoS

Enabling QoS Globally

By default, QoS is disabled on the switch.

The following procedure to enable QoS globally is required.

SUMMARY STEPS

1. **configure terminal**
2. **mls qos**
3. **end**
4. **show mls qos**
5. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode.
Step 2	mls qos Example: Device(config)# <code>mls qos</code>	Enables QoS globally. QoS operates with the default settings described in the related topic sections below. Note To disable QoS, use the no mls qos global configuration command.
Step 3	end Example: Device(config)# <code>end</code>	Returns to privileged EXEC mode.
Step 4	show mls qos Example: Device# <code>show mls qos</code>	Verifies the QoS configuration.
Step 5	copy running-config startup-config Example: Device# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Enabling VLAN-Based QoS on Physical Ports

By default, VLAN-based QoS is disabled on all physical switch ports. The switch applies QoS, including class maps and policy maps, only on a physical-port basis. You can enable VLAN-based QoS on a switch port.

The following procedure is required on physical ports that are specified in the interface level of a hierarchical policy map on a Switch Virtual Interface (SVI).

SUMMARY STEPS

1. `configure terminal`
2. `interface interface-id`
3. `mls qos vlan-based`
4. `end`

5. `show mls qos interface interface-id`
6. `copy running-config startup-config`

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode.
Step 2	interface interface-id Example: Device(config)# <code>interface gigabitethernet 1/0/1</code>	Specifies the physical port, and enter interface configuration mode.
Step 3	mls qos vlan-based Example: Device(config-if)# <code>mls qos vlan-based</code>	Enables VLAN-based QoS on the port. Note Use the <code>no mls qos vlan-based</code> interface configuration command to disable VLAN-based QoS on the physical port.
Step 4	end Example: Device(config-if)# <code>end</code>	Returns to privileged EXEC mode.
Step 5	show mls qos interface interface-id Example: Device# <code>show mls qos interface gigabitethernet 1/0/1</code>	Verifies if VLAN-based QoS is enabled on the physical port.
Step 6	copy running-config startup-config Example: Device# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Configuring Classification Using Port Trust States

These sections describe how to classify incoming traffic by using port trust states.

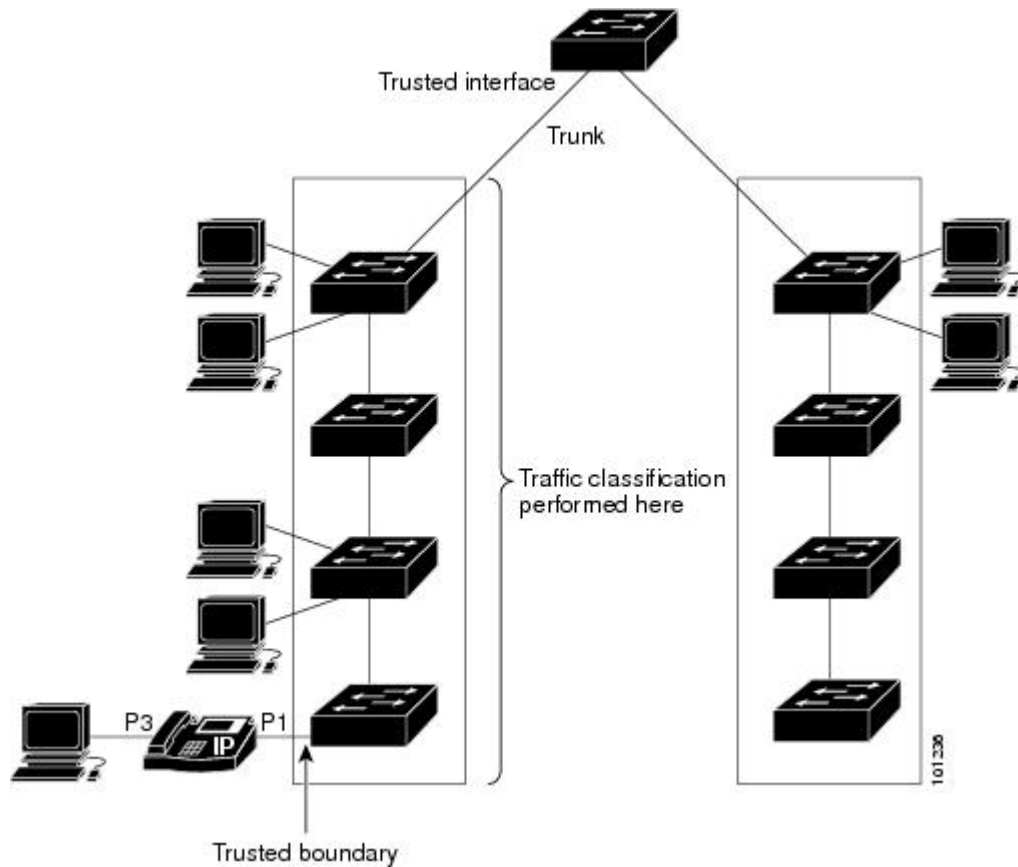


Note Depending on your network configuration, you must perform one or more of these tasks in this module or one or more of the tasks in the [Configuring a QoS Policy](#).

Configuring the Trust State on Ports Within the QoS Domain

Packets entering a QoS domain are classified at the edge of the QoS domain. When the packets are classified at the edge, the switch port within the QoS domain can be configured to one of the trusted states because there is no need to classify the packets at every switch within the QoS domain.

Figure 79: Port Trusted States on Ports Within the QoS Domain



SUMMARY STEPS

1. **configure terminal**
2. **interface *interface-id***
3. **mls qos trust [cos | dscp | ip-precedence]**
4. **end**
5. **show mls qos interface**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p>configure terminal</p> <p>Example:</p> <pre>Device# configure terminal</pre>	Enters global configuration mode.
Step 2	<p>interface <i>interface-id</i></p> <p>Example:</p> <pre>Device(config)# interface gigabitethernet 1/0/2</pre>	Specifies the port to be trusted, and enters interface configuration mode. Valid interfaces are physical ports.
Step 3	<p>mls qos trust [cos dscp ip-precedence]</p> <p>Example:</p> <pre>Device(config-if)# mls qos trust cos</pre>	<p>Configures the port trust state.</p> <p>By default, the port is not trusted. If no keyword is specified, the default is dscp.</p> <p>The keywords have these meanings:</p> <ul style="list-style-type: none"> • cos—Classifies an ingress packet by using the packet CoS value. For an untagged packet, the port default CoS value is used. The default port CoS value is 0. • dscp—Classifies an ingress packet by using the packet DSCP value. For a non-IP packet, the packet CoS value is used if the packet is tagged; for an untagged packet, the default port CoS is used. Internally, the switch maps the CoS value to a DSCP value by using the CoS-to-DSCP map. • ip-precedence—Classifies an ingress packet by using the packet IP-precedence value. For a non-IP packet, the packet CoS value is used if the packet is tagged; for an untagged packet, the default port CoS is used. Internally, the switch maps the CoS value to a DSCP value by using the CoS-to-DSCP map. <p>To return a port to its untrusted state, use the no mls qos trust interface configuration command.</p>
Step 4	<p>end</p> <p>Example:</p> <pre>Device(config-if)# end</pre>	Returns to privileged EXEC mode.
Step 5	<p>show mls qos interface</p> <p>Example:</p>	Verifies your entries.

	Command or Action	Purpose
	Device# <code>show mls qos interface</code>	
Step 6	copy running-config startup-config Example: Device# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Configuring the CoS Value for an Interface

QoS assigns the CoS value specified with the **mls qos cos** interface configuration command to untagged frames received on trusted and untrusted ports.

Beginning in privileged EXEC mode, follow these steps to define the default CoS value of a port or to assign the default CoS to all incoming packets on the port.

SUMMARY STEPS

1. **configure terminal**
2. **interface *interface-id***
3. **mls qos cos {*default-cos* | **override**}**
4. **end**
5. **show mls qos interface**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode.
Step 2	interface <i>interface-id</i> Example: Device(config)# <code>interface gigabitethernet 1/1/1</code>	Specifies the port to be configured, and enters interface configuration mode. Valid interfaces include physical ports.
Step 3	mls qos cos {<i>default-cos</i> override} Example: Device(config-if)# <code>mls qos</code>	Configures the default CoS value for the port. <ul style="list-style-type: none"> • For <i>default-cos</i>, specify a default CoS value to be assigned to a port. If the packet is untagged, the default CoS value becomes the packet CoS value. The CoS range is 0 to 7. The default is 0.

	Command or Action	Purpose
	<pre>override</pre>	<ul style="list-style-type: none"> Use the override keyword to override the previously configured trust state of the incoming packet and to apply the default port CoS value to the port on all incoming packets. By default, CoS override is disabled. <p>Use the override keyword when all incoming packets on specified ports deserve higher or lower priority than packets entering from other ports. Even if a port was previously set to trust DSCP, CoS, or IP precedence, this command overrides the previously configured trust state, and all the incoming CoS values are assigned the default CoS value configured with this command. If an incoming packet is tagged, the CoS value of the packet is modified with the default CoS of the port at the ingress port.</p> <p>Note To return to the default setting, use the no mls qos cos {default-cos override} interface configuration command.</p>
Step 4	<pre>end</pre> <p>Example:</p> <pre>Device(config-if)# end</pre>	Returns to privileged EXEC mode.
Step 5	<pre>show mls qos interface</pre> <p>Example:</p> <pre>Device# show mls qos interface</pre>	Verifies your entries.
Step 6	<pre>copy running-config startup-config</pre> <p>Example:</p> <pre>Device# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Configuring a Trusted Boundary to Ensure Port Security

In a typical network, you connect a Cisco IP Phone to a port and cascade devices that generate data packets from the back of the telephone. The Cisco IP Phone guarantees the voice quality through a shared data link by marking the CoS level of the voice packets as high priority (CoS = 5) and by marking the data packets as low priority (CoS = 0). Traffic sent from the telephone to the is typically marked with a tag that uses the 802.1Q header. The header contains the VLAN information and the class of service (CoS) 3-bit field, which is the priority of the packet.

For most Cisco IP Phone configurations, the traffic sent from the telephone to the should be trusted to ensure that voice traffic is properly prioritized over other types of traffic in the network. By using the **mls qos trust**

cos interface configuration command, you configure the port to which the telephone is connected to trust the CoS labels of all traffic received on that port. Use the **mls qos trust dscp** interface configuration command to configure a routed port to which the telephone is connected to trust the DSCP labels of all traffic received on that port.

With the trusted setting, you also can use the trusted boundary feature to prevent misuse of a high-priority queue if a user bypasses the telephone and connects the PC directly to the . Without trusted boundary, the CoS labels generated by the PC are trusted by the (because of the trusted CoS setting). By contrast, trusted boundary uses CDP to detect the presence of a Cisco IP Phone (such as the Cisco IP Phone 7910, 7935, 7940, and 7960) on a port. If the telephone is not detected, the trusted boundary feature disables the trusted setting on the port and prevents misuse of a high-priority queue. Note that the trusted boundary feature is not effective if the PC and Cisco IP Phone are connected to a hub that is connected to the .

In some situations, you can prevent a PC connected to the Cisco IP Phone from taking advantage of a high-priority data queue. You can use the **switchport priority extend cos** interface configuration command to configure the telephone through the CLI to override the priority of the traffic received from the PC.

SUMMARY STEPS

1. **configure terminal**
2. **cdp run**
3. **interface *interface-id***
4. **cdp enable**
5. Use one of the following:
 - **mls qos trust cos**
 - **mls qos trust dscp**
6. **mls qos trust device cisco-phone**
7. **end**
8. **show mls qos interface**
9. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 2	cdp run Example: Device(config)# cdp run	Enables CDP globally. By default, CDP is enabled.
Step 3	interface <i>interface-id</i> Example:	Specifies the port connected to the Cisco IP Phone, and enters interface configuration mode. Valid interfaces include physical ports.

	Command or Action	Purpose
	Device(config)# interface gigabitethernet 2/1/1	
Step 4	cdp enable Example: Device(config-if)# cdp enable	Enables CDP on the port. By default, CDP is enabled.
Step 5	Use one of the following: <ul style="list-style-type: none"> • mls qos trust cos • mls qos trust dscp Example: Device(config-if)# mls qos trust cos	Configures the port to trust the CoS value in traffic received from the Cisco IP Phone. or Configures the routed port to trust the DSCP value in traffic received from the Cisco IP Phone. By default, the port is not trusted.
Step 6	mls qos trust device cisco-phone Example: Device(config-if)# mls qos trust device cisco-phone	Specifies that the Cisco IP Phone is a trusted device. You cannot enable both trusted boundary and auto-QoS (auto qos voip interface configuration command) at the same time; they are mutually exclusive. Note To disable the trusted boundary feature, use the no mls qos trust device interface configuration command.
Step 7	end Example: Device(config-if)# end	Returns to privileged EXEC mode.
Step 8	show mls qos interface Example: Device# show mls qos interface	Verifies your entries.
Step 9	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Enabling DSCP Transparency Mode

The switch supports the DSCP transparency feature. It affects only the DSCP field of a packet at egress. By default, DSCP transparency is disabled. The switch modifies the DSCP field in an incoming packet, and the DSCP field in the outgoing packet is based on the quality of service (QoS) configuration, including the port trust setting, policing and marking, and the DSCP-to-DSCP mutation map.

If DSCP transparency is enabled by using the **no mls qos rewrite ip dscp** command, the switch does not modify the DSCP field in the incoming packet, and the DSCP field in the outgoing packet is the same as that in the incoming packet.

Regardless of the DSCP transparency configuration, the switch modifies the internal DSCP value of the packet, which the switch uses to generate a class of service (CoS) value that represents the priority of the traffic. The switch also uses the internal DSCP value to select an egress queue and threshold.

SUMMARY STEPS

1. **configure terminal**
2. **mls qos**
3. **no mls qos rewrite ip dscp**
4. **end**
5. **show mls qos interface** [*interface-id*]
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 2	mls qos Example: Device(config)# mls qos	Enables QoS globally.
Step 3	no mls qos rewrite ip dscp Example: Device(config)# no mls qos rewrite ip dscp	Enables DSCP transparency. The switch is configured to not modify the DSCP field of the IP packet.
Step 4	end Example: Device(config)# end	Returns to privileged EXEC mode.

	Command or Action	Purpose
Step 5	show mls qos interface [<i>interface-id</i>] Example: <pre>Device# show mls qos interface gigabitethernet 2/1/1</pre>	Verifies your entries.
Step 6	copy running-config startup-config Example: <pre>Device# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

DSCP Transparency Mode

To configure the switch to modify the DSCP value based on the trust setting or on an ACL by disabling DSCP transparency, use the **mls qos rewrite ip dscp** global configuration command.

If you disable QoS by using the **no mls qos** global configuration command, the CoS and DSCP values are not changed (the default QoS setting).

If you enter the **no mls qos rewrite ip dscp** global configuration command to enable DSCP transparency and then enter the **mls qos trust [cos | dscp]** interface configuration command, DSCP transparency is still enabled.

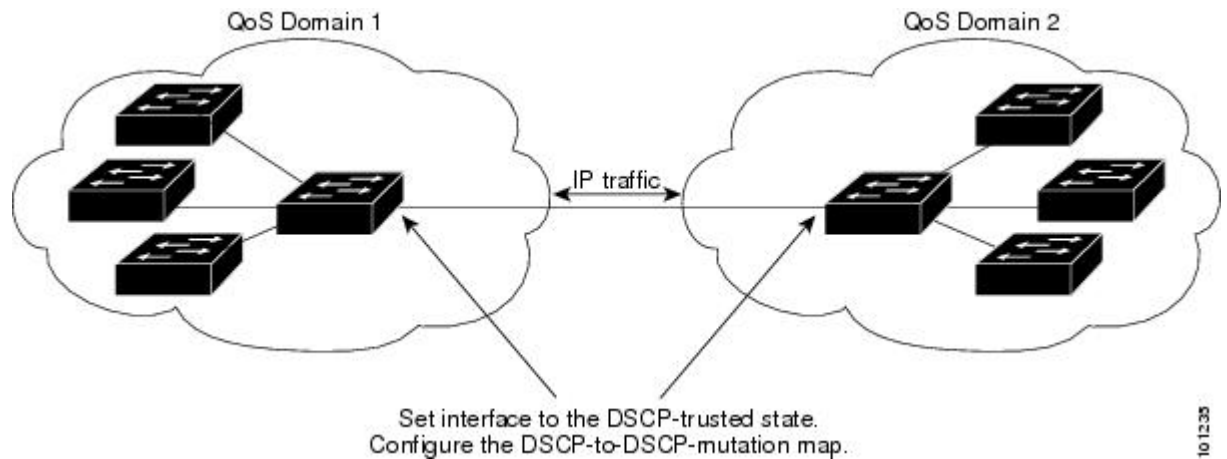


Note For Catalyst 2960-L switches, DSCP transparency is enabled by default.

Configuring the DSCP Trust State on a Port Bordering Another QoS Domain

If you are administering two separate QoS domains between which you want to implement QoS features for IP traffic, you can configure the ports bordering the domains to a DSCP-trusted state. The receiving port accepts the DSCP-trusted value and avoids the classification stage of QoS. If the two domains use different DSCP values, you can configure the DSCP-to-DSCP-mutation map to translate a set of DSCP values to match the definition in the other domain.

Figure 80: DSCP-Trusted State on a Port Bordering Another QoS Domain



10.13.35

Beginning in privileged EXEC mode, follow these steps to configure the DSCP-trusted state on a port and modify the DSCP-to-DSCP-mutation map. To ensure a consistent mapping strategy across both QoS domains, you must perform this procedure on the ports in both domains.

SUMMARY STEPS

1. **configure terminal**
2. **mls qos map dscp-mutation** *dscp-mutation-name in-dscp to out-dscp*
3. **interface** *interface-id*
4. **mls qos trust dscp**
5. **mls qos dscp-mutation** *dscp-mutation-name*
6. **end**
7. **show mls qos maps dscp-mutation**
8. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 2	mls qos map dscp-mutation <i>dscp-mutation-name in-dscp to out-dscp</i> Example: Device(config)# mls qos map dscp-mutation gigabitethernet1/0/2-mutation 10 11 12 13 to 30	Modifies the DSCP-to-DSCP-mutation map. The default DSCP-to-DSCP-mutation map is a null map, which maps an incoming DSCP value to the same DSCP value. <ul style="list-style-type: none"> • For <i>dscp-mutation-name</i>, enter the mutation map name. You can create more than one map by specifying a new name.

	Command or Action	Purpose
		<ul style="list-style-type: none"> For <i>in-dscp</i>, enter up to eight DSCP values separated by spaces. Then enter the to keyword. For <i>out-dscp</i>, enter a single DSCP value. <p>The DSCP range is 0 to 63.</p>
Step 3	interface <i>interface-id</i> Example: <pre>Device(config)# interface gigabitethernet1/0/2</pre>	<p>Specifies the port to be trusted, and enter interface configuration mode.</p> <p>Valid interfaces include physical ports.</p>
Step 4	mls qos trust dscp Example: <pre>Device(config-if)# mls qos trust dscp</pre>	<p>Configures the ingress port as a DSCP-trusted port. By default, the port is not trusted.</p> <p>Note To return a port to its non-trusted state, use the no mls qos trust interface configuration command.</p>
Step 5	mls qos dscp-mutation <i>dscp-mutation-name</i> Example: <pre>Device(config-if)# mls qos dscp-mutation gigabitethernet1/0/2-mutation</pre>	<p>Applies the map to the specified ingress DSCP-trusted port.</p> <p>For <i>dscp-mutation-name</i>, specify the mutation map name created in Step 2.</p> <p>You can configure multiple DSCP-to-DSCP-mutation maps on an ingress port.</p> <p>Note To return to the default DSCP-to-DSCP-mutation map values, use the no mls qos map dscp-mutation <i>dscp-mutation-name</i> global configuration command.</p>
Step 6	end Example: <pre>Device(config-if)# end</pre>	<p>Returns to privileged EXEC mode.</p>
Step 7	show mls qos maps dscp-mutation Example: <pre>Device# show mls qos maps dscp-mutation</pre>	<p>Verifies your entries.</p>
Step 8	copy running-config startup-config Example:	<p>(Optional) Saves your entries in the configuration file.</p>

	Command or Action	Purpose
	<pre>Device# copy-running-config startup-config</pre>	<p>Note To return a port to its non-trusted state, use the no mls qos trust interface configuration command. To return to the default DSCP-to-DSCP-mutation map values, use the no mls qos map dscp-mutation <i>dscp-mutation-name</i> global configuration command.</p>

Configuring a QoS Policy

Configuring a QoS policy typically requires the following tasks:

- Classifying traffic into classes
- Configuring policies applied to those traffic classes
- Attaching policies to ports

These sections describe how to classify, police, and mark traffic. Depending on your network configuration, you must perform one or more of the modules in this section.

Classifying Traffic by Using ACLs

You can classify IP traffic by using IPv4 standard ACLs, IPv4 extended ACLs, or IPv6 ACLs.

You can classify non-IP traffic by using Layer 2 MAC ACLs.

Creating an IP Standard ACL for IPv4 Traffic

Before you begin

Before you perform this task, determine which access lists you will be using for your QoS configuration.

SUMMARY STEPS

1. **configure terminal**
2. **access-list** *access-list-number* {deny | permit} *source* [*source-wildcard*]
3. **end**
4. **show access-lists**
5. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p>configure terminal</p> <p>Example:</p> <pre>Device# configure terminal</pre>	Enters global configuration mode.

	Command or Action	Purpose
Step 2	<p>access-list <i>access-list-number</i> {deny permit} <i>source</i> [<i>source-wildcard</i>]</p> <p>Example:</p> <pre>Device(config)# access-list 1 permit 192.2.255.0 10.1.1.255</pre>	<p>Creates an IP standard ACL, repeating the command as many times as necessary.</p> <ul style="list-style-type: none"> For <i>access-list-number</i>, enter the access list number. The range is 1 to 99 and 1300 to 1999. Use the permit keyword to permit a certain type of traffic if the conditions are matched. Use the deny keyword to deny a certain type of traffic if conditions are matched. For <i>source</i>, enter the network or host from which the packet is being sent. You can use the any keyword as an abbreviation for 0.0.0.0 255.255.255.255. (Optional) For <i>source-wildcard</i>, enter the wildcard bits in dotted decimal notation to be applied to the source. Place ones in the bit positions that you want to ignore. <p>When you create an access list, remember that by default the end of the access list contains an implicit deny statement for everything if it did not find a match before reaching the end.</p> <p>Note To delete an access list, use the no access-list <i>access-list-number</i> global configuration command.</p>
Step 3	<p>end</p> <p>Example:</p> <pre>Device(config)# end</pre>	Returns to privileged EXEC mode.
Step 4	<p>show access-lists</p> <p>Example:</p> <pre>Device# show access-lists</pre>	Verifies your entries.
Step 5	<p>copy running-config startup-config</p> <p>Example:</p> <pre>Device# copy-running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Creating an IP Extended ACL for IPv4 Traffic

Before you begin

Before you perform this task, determine which access lists you will be using for your QoS configuration.

SUMMARY STEPS

1. **configure terminal**
2. **access-list** *access-list-number* {**deny** | **permit**} *protocol source source-wildcard destination destination-wildcard*
3. **end**
4. **show access-lists**
5. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 2	access-list <i>access-list-number</i> { deny permit } <i>protocol source source-wildcard destination destination-wildcard</i> Example: Device(config)# access-list 100 permit ip any any dscp 32	Creates an IP extended ACL, repeating the command as many times as necessary. <ul style="list-style-type: none"> • For <i>access-list-number</i>, enter the access list number. The range is 100 to 199 and 2000 to 2699. • Use the permit keyword to permit a certain type of traffic if the conditions are matched. Use the deny keyword to deny a certain type of traffic if conditions are matched. • For <i>protocol</i>, enter the name or number of an IP protocol. Use the question mark (?) to see a list of available protocol keywords. • For <i>source</i>, enter the network or host from which the packet is being sent. You specify this by using dotted decimal notation, by using the any keyword as an abbreviation for <i>source 0.0.0.0 source-wildcard 255.255.255.255</i>, or by using the host keyword for <i>source 0.0.0.0</i>. • For <i>source-wildcard</i>, enter the wildcard bits by placing ones in the bit positions that you want to ignore. You specify the wildcard by using dotted decimal notation, by using the any keyword as an abbreviation for <i>source 0.0.0.0 source-wildcard 255.255.255.255</i>, or by using the host keyword for <i>source 0.0.0.0</i>.

	Command or Action	Purpose
		<ul style="list-style-type: none"> For <i>destination</i>, enter the network or host to which the packet is being sent. You have the same options for specifying the <i>destination and destination-wildcard</i> as those described by <i>source</i> and <i>source-wildcard</i>. <p>When creating an access list, remember that, by default, the end of the access list contains an implicit deny statement for everything if it did not find a match before reaching the end.</p> <p>Note To delete an access list, use the no access-list access-list-number global configuration command.</p>
Step 3	end Example: Device(config)# end	Returns to privileged EXEC mode.
Step 4	show access-lists Example: Device# show access-lists	Verifies your entries.
Step 5	copy running-config startup-config Example: Device# copy-running-config startup-config	(Optional) Saves your entries in the configuration file.

Creating an IPv6 ACL for IPv6 Traffic

Before you begin

Before you perform this task, determine which access lists you will be using for your QoS configuration.

SUMMARY STEPS

1. **configure terminal**
2. **ipv6 access-list access-list-name**
3. **{deny | permit} protocol {source-ipv6-prefix/prefix-length | any | host source-ipv6-address} [operator [port-number]] {destination-ipv6-prefix/ prefix-length | any | host destination-ipv6-address} [operator [port-number]] [dscp value] [fragments] [log] [log-input] [routing] [sequence value] [time-range name]**
4. **end**
5. **show ipv6 access-list**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>Device# configure terminal</pre>	Enters global configuration mode.
Step 2	ipv6 access-list <i>access-list-name</i> Example: <pre>Device(config)# ipv6 access-list ipv6_Name_ACL</pre>	<p>Creates an IPv6 ACL and enters IPv6 access-list configuration mode.</p> <p>Accesses list names cannot contain a space or quotation mark or begin with a numeric.</p> <p>Note To delete an access list, use the no ipv6 access-list <i>access-list-number</i> global configuration command.</p>
Step 3	<pre>{deny permit} protocol {source-ipv6-prefix/prefix-length any host source-ipv6-address} [operator [port-number]] {destination-ipv6-prefix/ prefix-length any host destination-ipv6-address} [operator [port-number]] [dscp value] [fragments] [log] [log-input] [routing] [sequence value] [time-range name]</pre> Example: <pre>Device(config-ipv6-acl)# permit ip host 10::1 host 11::2 host</pre>	<p>Enters deny or permit to specify whether to deny or permit the packet if conditions are matched. These are the conditions:</p> <p>For <i>protocol</i>, enter the name or number of an Internet protocol: ahp, esp, icmp, ipv6, pcp, stcp, tcp, or udp, or an integer in the range 0 to 255 representing an IPv6 protocol number.</p> <ul style="list-style-type: none"> The <i>source-ipv6-prefix/prefix-length</i> or <i>destination-ipv6-prefix/ prefix-length</i> is the source or destination IPv6 network or class of networks for which to set deny or permit conditions, specified in hexadecimal and using 16-bit values between colons (see RFC 2373). Enter any as an abbreviation for the IPv6 prefix <code>::/0</code>. For host <i>source-ipv6-address</i> or destination-ipv6-address, enter the source or destination IPv6 host address for which to set deny or permit conditions, specified in hexadecimal using 16-bit values between colons. (Optional) For <i>operator</i>, specify an operand that compares the source or destination ports of the specified protocol. Operands are lt (less than), gt (greater than), eq (equal), neq (not equal), and range. <p>If the operator follows the <i>source-ipv6-prefix/prefix-length</i> argument, it must match the source port. If the operator follows the <i>destination-ipv6- prefix/prefix-length</i> argument, it must match the destination port.</p>

	Command or Action	Purpose
		<ul style="list-style-type: none"> • (Optional) The <i>port-number</i> is a decimal number from 0 to 65535 or the name of a TCP or UDP port. You can use TCP port names only when filtering TCP. You can use UDP port names only when filtering UDP. • (Optional) Enter dscp value to match a differentiated services code point value against the traffic class value in the Traffic Class field of each IPv6 packet header. The acceptable range is from 0 to 63. • (Optional) Enter fragments to check noninitial fragments. This keyword is visible only if the protocol is IPv6. • (Optional) Enter log to cause a logging message to be sent to the console about the packet that matches the entry. Enter log-input to include the input interface in the log entry. Logging is supported only for router ACLs. • (Optional) Enter routing to specify that IPv6 packets be routed. • (Optional) Enter sequence value to specify the sequence number for the access list statement. The acceptable range is from 1 to 4294967295. • (Optional) Enter time-range name to specify the time range that applies to the deny or permit statement.
Step 4	end Example: <pre>Device(config-ipv6-acl)# end</pre>	Returns to privileged EXEC mode.
Step 5	show ipv6 access-list Example: <pre>Device# show ipv6 access-list</pre>	Verifies the access list configuration.
Step 6	copy running-config startup-config Example: <pre>Device# copy-running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Creating a Layer 2 MAC ACL for Non-IP Traffic

Before you begin

Before you perform this task, determine that Layer 2 MAC access lists are required for your QoS configuration.

SUMMARY STEPS

1. **configure terminal**
2. **mac access-list extended name**
3. **{permit | deny} {host src-MAC-addr mask | any | host dst-MAC-addr | dst-MAC-addr mask} [type mask]**
4. **end**
5. **show access-lists [access-list-number | access-list-name]**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 2	mac access-list extended name Example: Device(config)# mac access-list extended maclist1	Creates a Layer 2 MAC ACL by specifying the name of the list. After entering this command, the mode changes to extended MAC ACL configuration. Note To delete an access list, use the no mac access-list extended access-list-name global configuration command.
Step 3	{permit deny} {host src-MAC-addr mask any host dst-MAC-addr dst-MAC-addr mask} [type mask] Example: Device(config-ext-macl) # permit 0001.0000.0001 0.0.0 0002.0000.0001 0.0.0 Device(config-ext-macl) # permit 0001.0000.0002 0.0.0 0002.0000.0002 0.0.0 xns-idp	Specifies the type of traffic to permit or deny if the conditions are matched, entering the command as many times as necessary. <ul style="list-style-type: none"> • For <i>src-MAC-addr</i>, enter the MAC address of the host from which the packet is being sent. You specify this by using the hexadecimal format (H.H.H), by using the any keyword as an abbreviation for <i>source</i> 0.0.0, <i>source-wildcard</i> ffff.ffff.ffff, or by using the host keyword for <i>source</i> 0.0.0. • For <i>mask</i>, enter the wildcard bits by placing ones in the bit positions that you want to ignore. • For <i>dst-MAC-addr</i>, enter the MAC address of the host to which the packet is being sent. You specify this by using the hexadecimal format (H.H.H), by using the any keyword as an abbreviation for <i>source</i> 0.0.0,

	Command or Action	Purpose
		<p><i>source-wildcard</i> ffff.ffff.ffff, or by using the host keyword for <i>source</i> 0.0.0.</p> <ul style="list-style-type: none"> (Optional) For <i>type mask</i>, specify the Ethertype number of a packet with Ethernet II or SNAP encapsulation to identify the protocol of the packet. For <i>type</i>, the range is from 0 to 65535, typically specified in hexadecimal. For <i>mask</i>, enter the <i>don't care</i> bits applied to the Ethertype before testing for a match. <p>When creating an access list, remember that, by default, the end of the access list contains an implicit deny statement for everything if it did not find a match before reaching the end.</p>
Step 4	<p>end</p> <p>Example:</p> <pre>Device(config-ext-macl)# end</pre>	Returns to privileged EXEC mode.
Step 5	<p>show access-lists [<i>access-list-number</i> <i>access-list-name</i>]</p> <p>Example:</p> <pre>Device# show access-lists</pre>	Verifies your entries.
Step 6	<p>copy running-config startup-config</p> <p>Example:</p> <pre>Device# copy-running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Classifying Traffic by Using Class Maps

You use the **class-map** global configuration command to name and to isolate a specific traffic flow (or class) from all other traffic. The class map defines the criteria to use to match against a specific traffic flow to further classify it. Match statements can include criteria such as an ACL, IP precedence values, or DSCP values. The match criterion is defined with one match statement entered within the class-map configuration mode.



Note You can also create class maps during policy map creation by using the **class** policy-map configuration command.

SUMMARY STEPS

1. **configure terminal**

2. Use one of the following:

- **access-list** *access-list-number* {deny | permit} *source* [*source-wildcard*]
- **access-list** *access-list-number* {deny | permit} *protocol source* [*source-wildcard*] *destination* [*destination-wildcard*]
- **ipv6 access-list** *access-list-name* {deny | permit} *protocol* {*source-ipv6-prefix/prefix-length* | any | host *source-ipv6-address*} [*operator* [*port-number*]] {*destination-ipv6-prefix/ prefix-length* | any | host *destination-ipv6-address*} [*operator* [*port-number*]] [*dscp value*] [*fragments*] [*log*] [*log-input*] [*routing*] [*sequence value*] [*time-range name*]
- **mac access-list extended** *name* {permit | deny} {host *src-MAC-addr mask* | any | host *dst-MAC-addr* | *dst-MAC-addr mask*} [*type mask*]

3. **class-map** [match-all | match-any] *class-map-name*4. **match** {access-group *acl-index-or-name* | ip **dscp** *dscp-list* | ip **precedence** *ip-precedence-list*}5. **end**6. **show class-map**7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 2	Use one of the following: <ul style="list-style-type: none"> • access-list <i>access-list-number</i> {deny permit} <i>source</i> [<i>source-wildcard</i>] • access-list <i>access-list-number</i> {deny permit} <i>protocol source</i> [<i>source-wildcard</i>] <i>destination</i> [<i>destination-wildcard</i>] • ipv6 access-list <i>access-list-name</i> {deny permit} <i>protocol</i> {<i>source-ipv6-prefix/prefix-length</i> any host <i>source-ipv6-address</i>} [<i>operator</i> [<i>port-number</i>]] {<i>destination-ipv6-prefix/ prefix-length</i> any host <i>destination-ipv6-address</i>} [<i>operator</i> [<i>port-number</i>]] [<i>dscp value</i>] [<i>fragments</i>] [<i>log</i>] [<i>log-input</i>] [<i>routing</i>] [<i>sequence value</i>] [<i>time-range name</i>] • mac access-list extended <i>name</i> {permit deny} {host <i>src-MAC-addr mask</i> any host <i>dst-MAC-addr</i> <i>dst-MAC-addr mask</i>} [<i>type mask</i>] Example: Device(config)# access-list 103 permit ip any any dscp 10	Creates an IP standard or extended ACL, an IPv6 ACL for IP traffic, or a Layer 2 MAC ACL for non-IP traffic, repeating the command as many times as necessary. When creating an access list, remember that, by default, the end of the access list contains an implicit deny statement for everything if it did not find a match before reaching the end.

	Command or Action	Purpose
Step 3	<p>class-map [match-all match-any] <i>class-map-name</i></p> <p>Example:</p> <pre>Device(config)# class-map class1</pre>	<p>Creates a class map, and enters class-map configuration mode.</p> <p>By default, no class maps are defined.</p> <ul style="list-style-type: none"> • (Optional) Use the match-all keyword to perform a logical-AND of all matching statements under this class map. All match criteria in the class map must be matched. • (Optional) Use the match-any keyword to perform a logical-OR of all matching statements under this class map. One or more match criteria must be matched. • For <i>class-map-name</i>, specify the name of the class map. <p>If neither the match-all or match-any keyword is specified, the default is match-all.</p> <p>Note To delete an existing class map, use the no class-map [match-all match-any] class-map-name global configuration command.</p>
Step 4	<p>match {access-group <i>acl-index-or-name</i> ip dscp <i>dscp-list</i> ip precedence <i>ip-precedence-list</i>}</p> <p>Example:</p> <pre>Device(config-cmap)# match ip dscp 10 11 12</pre>	<p>Defines the match criterion to classify traffic.</p> <p>By default, no match criterion is defined.</p> <p>Only one match criterion per class map is supported, and only one ACL per class map is supported.</p> <ul style="list-style-type: none"> • For access-group <i>acl-index-or-name</i>, specify the number or name of the ACL created in Step 2. • To filter IPv6 traffic with the match access-group command, create an IPv6 ACL, as described in Step 2. • For ip dscp <i>dscp-list</i>, enter a list of up to eight IP DSCP values to match against incoming packets. Separate each value with a space. The range is 0 to 63. • For ip precedence <i>ip-precedence-list</i>, enter a list of up to eight IP-precedence values to match against incoming packets. Separate each value with a space. The range is 0 to 7. <p>Note To remove a match criterion, use the no match {access-group <i>acl-index-or-name</i> ip dscp ip precedence} class-map configuration command.</p>
Step 5	<p>end</p> <p>Example:</p>	<p>Returns to privileged EXEC mode.</p>

	Command or Action	Purpose
	Device(config-cmap) # end	
Step 6	show class-map Example: Device# show class-map	Verifies your entries.
Step 7	copy running-config startup-config Example: Device# copy-running-config startup-config	(Optional) Saves your entries in the configuration file.

Classifying Traffic by Using Class Maps and Filtering IPv6 Traffic

To apply the primary match criteria to only IPv4 traffic, use the **match protocol** command with the **ip** keyword. To apply the primary match criteria to only IPv6 traffic, use the **match protocol** command with the **ipv6** keyword.

SUMMARY STEPS

1. **configure terminal**
2. **class-map {match-all} class-map-name**
3. **match protocol [ip / ipv6]**
4. **match {ip dscp dscp-list | ip precedence ip-precedence-list}**
5. **end**
6. **show class-map**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 2	class-map {match-all} class-map-name Example: Device(config) # class-map cm-1	Creates a class map, and enters class-map configuration mode. By default, no class maps are defined. When you use the match protocol command, only the match-all keyword is supported.

	Command or Action	Purpose
		<ul style="list-style-type: none"> For <i>class-map-name</i>, specify the name of the class map. <p>If neither the match-all or match-any keyword is specified, the default is match-all.</p> <p>Note To delete an existing class map, use the no class-map [match-all match-any] class-map-name global configuration command.</p>
Step 3	<p>match protocol [<i>ip / ipv6</i>]</p> <p>Example:</p> <pre>Device(config-cmap)# match protocol ip</pre>	<p>(Optional) Specifies the IP protocol to which the class map applies:</p> <ul style="list-style-type: none"> Use the argument <i>ip</i> to specify IPv4 traffic and <i>ipv6</i> to specify IPv6 traffic. When you use the match protocol command, only the match-all keyword is supported for the class-map command. <p>Note You can also match the following protocols: ARP, CDP, and HTTP.</p>
Step 4	<p>match {ip dscp dscp-list ip precedence ip-precedence-list}</p> <p>Example:</p> <pre>Device(config-cmap)# match ip dscp 10</pre>	<p>Defines the match criterion to classify traffic.</p> <p>By default, no match criterion is defined.</p> <ul style="list-style-type: none"> For ip dscp dscp-list, enter a list of up to eight IP DSCP values to match against incoming packets. Separate each value with a space. The range is 0 to 63. For ip precedence ip-precedence-list, enter a list of up to eight IP-precedence values to match against incoming packets. Separate each value with a space. The range is 0 to 7. <p>Note To remove a match criterion, use the no match {access-group acl-index-or-name ip dscp ip precedence} class-map configuration command.</p>
Step 5	<p>end</p> <p>Example:</p> <pre>Device(config-cmap)# end</pre>	<p>Returns to privileged EXEC mode.</p>
Step 6	<p>show class-map</p> <p>Example:</p> <pre>Device# show class-map</pre>	<p>Verifies your entries.</p>

	Command or Action	Purpose
Step 7	copy running-config startup-config Example: Device# <code>copy-running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Classifying, Policing, and Marking Traffic on Physical Ports by Using Policy Maps

You can configure a policy map on a physical port that specifies which traffic class to act on. Actions can include trusting the CoS, DSCP, or IP precedence values in the traffic class; setting a specific DSCP or IP precedence value in the traffic class; and specifying the traffic bandwidth limitations for each matched traffic class (policer) and the action to take when the traffic is out of profile (marking).

A policy map also has these characteristics:

- A policy map can contain multiple class statements, each with different match criteria and policers.
- A policy map can contain a predefined default traffic class explicitly placed at the end of the map.
- A separate policy-map class can exist for each type of traffic received through a port.

Follow these guidelines when configuring policy maps on physical ports:

- You can attach only one policy map per ingress port.
- If you configure the IP-precedence-to-DSCP map by using the **mls qos map ip-prec-dscp dscp1...dscp8** global configuration command, the settings only affect packets on ingress interfaces that are configured to trust the IP precedence value. In a policy map, if you set the packet IP precedence value to a new value by using the **set ip precedence new-precedence** policy-map class configuration command, the egress DSCP value is not affected by the IP-precedence-to-DSCP map. If you want the egress DSCP value to be different than the ingress value, use the **set dscp new-dscp** policy-map class configuration command.
- If you enter or have used the **set ip dscp** command, the changes this command to **set dscp** in its configuration.
- You can use the **set ip precedence** or the **set precedence** policy-map class configuration command to change the packet IP precedence value. This setting appears as **set ip precedence** in the configuration.
- You can configure a separate second-level policy map for each class defined for the port. The second-level policy map specifies the police action to take for each traffic class.
- A policy-map and a port trust state can both run on a physical interface. The policy-map is applied before the port trust state.
- When you configure a default traffic class by using the **class class-default** policy-map configuration command, unclassified traffic (traffic that does not meet the match criteria specified in the traffic classes) is treated as the default traffic class (**class-default**).

SUMMARY STEPS

1. **configure terminal**
2. **class-map [match-all | match-any] class-map-name**
3. **policy-map policy-map-name**

4. **class** [*class-map-name* | **class-default**]
5. **trust** [**cos** | **dscp** | **ip-precedence**]
6. **set** {**dscp** *new-dscp* | **ip precedence** *new-precedence*}
7. **police** *rate-bps burst-byte* [**exceed-action** {**drop** | **policed-dscp-transmit**}]
8. **exit**
9. **exit**
10. **interface** *interface-id*
11. **service-policy input** *policy-map-name*
12. **end**
13. **show policy-map** [*policy-map-name* [**class** *class-map-name*]]
14. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p>configure terminal</p> <p>Example:</p> <pre>Device# configure terminal</pre>	Enters global configuration mode.
Step 2	<p>class-map [match-all match-any] <i>class-map-name</i></p> <p>Example:</p> <pre>Device(config)# class-map ipclass1</pre>	<p>Creates a class map, and enters class-map configuration mode.</p> <p>By default, no class maps are defined.</p> <ul style="list-style-type: none"> • (Optional) Use the match-all keyword to perform a logical-AND of all matching statements under this class map. All match criteria in the class map must be matched. • (Optional) Use the match-any keyword to perform a logical-OR of all matching statements under this class map. One or more match criteria must be matched. • For <i>class-map-name</i>, specify the name of the class map. <p>If neither the match-all or match-any keyword is specified, the default is match-all.</p>
Step 3	<p>policy-map <i>policy-map-name</i></p> <p>Example:</p> <pre>Device(config-cmap)# policy-map flowit</pre>	<p>Creates a policy map by entering the policy map name, and enters policy-map configuration mode.</p> <p>By default, no policy maps are defined.</p> <p>The default behavior of a policy map is to set the DSCP to 0 if the packet is an IP packet and to set the CoS to 0 if the packet is tagged. No policing is performed.</p>

	Command or Action	Purpose
		<p>Note To delete an existing policy map, use the no policy-map <i>policy-map-name</i> global configuration command.</p>
Step 4	<p>class [<i>class-map-name</i> class-default]</p> <p>Example:</p> <pre>Device(config-pmap)# class ipclass1</pre>	<p>Defines a traffic classification, and enters policy-map class configuration mode.</p> <p>By default, no policy map class-maps are defined.</p> <p>If a traffic class has already been defined by using the class-map global configuration command, specify its name for <i>class-map-name</i> in this command.</p> <p>A class-default traffic class is pre-defined and can be added to any policy. It is always placed at the end of a policy map. With an implied match any included in the class-default class, all packets that have not already matched the other traffic classes will match class-default.</p> <p>Note To delete an existing class map, use the no class <i>class-map-name</i> policy-map configuration command.</p>
Step 5	<p>trust [cos dscp ip-precedence]</p> <p>Example:</p> <pre>Device(config-pmap-c)# trust dscp</pre>	<p>Configures the trust state, which QoS uses to generate a CoS-based or DSCP-based QoS label.</p> <p>This command is mutually exclusive with the set command within the same policy map. If you enter the trust command, go to Step 6.</p> <p>By default, the port is not trusted. If no keyword is specified when the command is entered, the default is dscp.</p> <p>The keywords have these meanings:</p> <ul style="list-style-type: none"> • cos—QoS derives the DSCP value by using the received or default port CoS value and the CoS-to-DSCP map. • dscp—QoS derives the DSCP value by using the DSCP value from the ingress packet. For non-IP packets that are tagged, QoS derives the DSCP value by using the received CoS value; for non-IP packets that are untagged, QoS derives the DSCP value by using the default port CoS value. In either case, the DSCP value is derived from the CoS-to-DSCP map. • ip-precedence—QoS derives the DSCP value by using the IP precedence value from the ingress packet and the IP-precedence-to-DSCP map. For non-IP packets that are tagged, QoS derives the DSCP value by using the received CoS value; for non-IP packets that are untagged, QoS derives the DSCP value by

	Command or Action	Purpose
		<p>using the default port CoS value. In either case, the DSCP value is derived from the CoS-to-DSCP map.</p> <p>Note To return to the untrusted state, use the no trust policy-map configuration command</p>
Step 6	<p>set {dscp <i>new-dscp</i> ip precedence <i>new-precedence</i>}</p> <p>Example:</p> <pre>Device(config-pmap-c)# set dscp 45</pre>	<p>Classifies IP traffic by setting a new value in the packet.</p> <ul style="list-style-type: none"> For dscp <i>new-dscp</i>, enter a new DSCP value to be assigned to the classified traffic. The range is 0 to 63. For ip precedence <i>new-precedence</i>, enter a new IP-precedence value to be assigned to the classified traffic. The range is 0 to 7. <p>Note To remove an assigned DSCP or IP precedence value, use the no set {dscp <i>new-dscp</i> ip precedence <i>new-precedence</i>} policy-map configuration command.</p>
Step 7	<p>police <i>rate-bps burst-byte</i> [exceed-action {drop policed-dscp-transmit}]</p> <p>Example:</p> <pre>Device(config-pmap-c)# police 100000 80000 drop</pre>	<p>Defines a policer for the classified traffic.</p> <p>By default, no policer is defined.</p> <ul style="list-style-type: none"> For <i>rate-bps</i>, specify average traffic rate in bits per second (b/s). The range is 8000 to 10000000000. For <i>burst-byte</i>, specify the normal burst size in bytes. The range is 8000 to 1000000. (Optional) Specifies the action to take when the rates are exceeded. Use the exceed-action drop keywords to drop the packet. Use the exceed-action policed-dscp-transmit keywords to mark down the DSCP value (by using the policed-DSCP map) and to send the packet. <p>Note To remove an existing policer, use the no police <i>rate-bps burst-byte</i> [exceed-action {drop policed-dscp-transmit}] policy-map configuration command.</p>
Step 8	<p>exit</p> <p>Example:</p> <pre>Device(config-pmap-c)# exit</pre>	Returns to policy map configuration mode.
Step 9	<p>exit</p> <p>Example:</p>	Returns to global configuration mode.

	Command or Action	Purpose
	Device (config-pmap) # exit	
Step 10	interface <i>interface-id</i> Example: Device (config) # interface gigabitethernet 2/0/1	Specifies the port to attach to the policy map, and enters interface configuration mode. Valid interfaces include physical ports.
Step 11	service-policy input <i>policy-map-name</i> Example: Device (config-if) # service-policy input flowit	Specifies the policy-map name, and applies it to an ingress port. Only one policy map per ingress port is supported. Note To remove the policy map and port association, use the no service-policy input <i>policy-map-name</i> interface configuration command.
Step 12	end Example: Device (config-if) # end	Returns to privileged EXEC mode.
Step 13	show policy-map [<i>policy-map-name</i> [class <i>class-map-name</i>]] Example: Device# show policy-map	Verifies your entries.
Step 14	copy running-config startup-config Example: Device# copy-running-config startup-config	(Optional) Saves your entries in the configuration file.

Classifying, Policing, and Marking Traffic on SVIs by Using Hierarchical Policy Maps

You can configure hierarchical policy maps on SVIs, but not on other types of interfaces. Hierarchical policing combines the VLAN- and interface-level policy maps to create a single policy map.

You can configure hierarchical policy maps on SVIs, but not on other types of interfaces. Hierarchical policing combines the VLAN- and interface-level policy maps to create a single policy map.

On an SVI, the VLAN-level policy map specifies which traffic class to act on. Actions can include trusting the CoS, DSCP, or IP precedence values or setting a specific DSCP or IP precedence value in the traffic class. Use the interface-level policy map to specify the physical ports that are affected by individual policers.

You can configure hierarchical policy maps that filter IPv4 and IPv6 traffic.

Follow these guidelines when configuring hierarchical policy maps:

- Before configuring a hierarchical policy map, you must enable VLAN-based QoS on the physical ports that are to be specified at the interface level of the policy map.
- You can attach only one policy map per ingress port or SVI.
- A policy map can contain multiple class statements, each with different match criteria and actions.
- A separate policy-map class can exist for each type of traffic received on the SVI.
- In a switch stack, you cannot use the **match input-interface** class-map configuration command to specify interfaces across stack members in a policy-map class.
- A policy-map and a port trust state can both run on a physical interface. The policy-map is applied before the port trust state.
- If you configure the IP-precedence-to-DSCP map by using the **mls qos map ip-prec-dscp dscp1...dscp8** global configuration command, the settings only affect packets on ingress interfaces that are configured to trust the IP precedence value. In a policy map, if you set the packet IP precedence value to a new value by using the **set ip precedence new-precedence** policy-map class configuration command, the egress DSCP value is not affected by the IP-precedence-to-DSCP map. If you want the egress DSCP value to be different than the ingress value, use the **set dscp new-dscp** policy-map class configuration command.
- If you enter or have used the **set ip dscp** command, the switch changes this command to **set dscp** in its configuration. If you enter the **set ip dscp** command, this setting appears as **set dscp** in the switch configuration.
- You can use the **set ip precedence** or the **set precedence** policy-map class configuration command to change the packet IP precedence value. This setting appears as **set ip precedence** in the switch configuration.
- If VLAN-based QoS is enabled, the hierarchical policy map supersedes the previously configured port-based policy map.
- The hierarchical policy map is attached to the SVI and affects all traffic in the VLAN. The actions specified in the VLAN-level policy map affect the traffic belonging to the SVI. The police action on the port-level policy map affects the ingress traffic on the affected physical interfaces.
- When configuring a hierarchical policy map on trunk ports, the VLAN ranges must not overlap. If the ranges overlap, the actions specified in the policy map affect the incoming and outgoing traffic on the overlapped VLANs.
- Aggregate policers are not supported in hierarchical policy maps.
- When VLAN-based QoS is enabled, the switch supports VLAN-based features, such as the VLAN map.
- You can configure a hierarchical policy map only on the primary VLAN of a private VLAN.
- When you enable VLAN-based QoS and configure a hierarchical policy map in a switch stack, these automatic actions occur when the stack configuration changes:
 - When a new active stack is selected, the active stack reenables and reconfigures these features on all applicable interfaces on the active stack.
 - When a stack member is added, the active stack reenables and reconfigures these features on all applicable ports on the stack member.

- When you merge switch stacks, the new active stack reenables and reconfigures these features on the switches in the new stack.
- When the switch stack divides into two or more switch stacks, the active stack in each switch stack re-enables and reconfigures these features on all applicable interfaces on the stack members, including the active stack.
- When you configure a default traffic class by using the **class class-default** policy-map configuration command, unclassified traffic (traffic that does not meet the match criteria specified in the traffic classes) is treated as default traffic class (**class-default**).

SUMMARY STEPS

1. **configure terminal**
2. **class-map** [**match-all** | **match-any**] *class-map-name*
3. **match** {**access-group** *acl-index-or-name* | **ip dscp** *dscp-list* | **ip precedence** *ip-precedence-list*}
4. **match protocol** [*ip* / *ipv6*]
5. **exit**
6. **exit**
7. **class-map** [**match-all** | **match-any**] *class-map-name*
8. **match input-interface** *interface-id-list*
9. **exit**
10. **exit**
11. **policy-map** *policy-map-name*
12. **class-map** *class-map-name*
13. **police** *rate-bps burst-byte* [**exceed-action** {**drop** | **policed-dscp-transmit**}]
14. **exit**
15. **exit**
16. **policy-map** *policy-map-name*
17. **class** [*class-map-name* | **class-default**]
18. **trust** [**cos** | **dscp** | **ip-precedence**]
19. **set** {**dscp** *new-dscp* | **ip precedence** *new-precedence*}
20. **service-policy** *policy-map-name*
21. **exit**
22. **exit**
23. **interface** *interface-id*
24. **service-policy input** *policy-map-name*
25. **end**
26. **show policy-map** [*policy-map-name* [**class** *class-map-name*]] or **show mls qos vlan-based**
27. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example:	Enters global configuration mode.

	Command or Action	Purpose
	Device# <code>configure terminal</code>	
Step 2	<p>class-map [match-all match-any] <i>class-map-name</i></p> <p>Example:</p> <pre>Device(config)# class-map cm-1</pre>	<p>Creates a VLAN-level class map, and enters class-map configuration mode.</p> <p>By default, no class maps are defined.</p> <ul style="list-style-type: none"> • (Optional) Use the match-all keyword to perform a logical-AND of all matching statements under this class map. All match criteria in the class map must be matched. • (Optional) Use the match-any keyword to perform a logical-OR of all matching statements under this class map. One or more match criteria must be matched. • For <i>class-map-name</i>, specify the name of the class map. <p>If neither the match-all or match-any keyword is specified, the default is match-all.</p> <p>Because only one match command per class map is supported, the match-all and match-any keywords function the same.</p>
Step 3	<p>match {access-group <i>acl-index-or-name</i> ip dscp <i>dscp-list</i> ip precedence <i>ip-precedence-list</i>}</p> <p>Example:</p> <pre>Device(config-cmap)# match ip dscp 10</pre>	<p>Defines the match criterion to classify traffic.</p> <p>By default, no match criterion is defined.</p> <p>Only one match criterion per class map is supported, and only one ACL per class map is supported.</p> <ul style="list-style-type: none"> • For access-group <i>acl-index-or-name</i>, specify the number or name of the ACL. • For ip dscp <i>dscp-list</i>, enter a list of up to eight IP DSCP values to match against incoming packets. Separate each value with a space. The range is 0 to 63. • For ip precedence <i>ip-precedence-list</i>, enter a list of up to eight IP-precedence values to match against incoming packets. Separate each value with a space. The range is 0 to 7.
Step 4	<p>match protocol [<i>ip</i> <i>ipv6</i>]</p> <p>Example:</p> <pre>Device(config-cmap)# match protocol ipv6</pre>	<p>(Optional) Specifies the IP protocol to which the class map applies.</p> <ul style="list-style-type: none"> • Use the argument <i>ip</i> to specify IPv4 traffic, and <i>ipv6</i> to specify IPv6 traffic.

	Command or Action	Purpose
		<ul style="list-style-type: none"> When you use the match protocol command, only the match-all keyword is supported for the first level class map. <p>You can use the match protocol command with the match ip dscp or match precedence commands, but not with the match access-group command.</p>
Step 5	exit Example: <pre>Device(config-cmap)# exit</pre>	Returns to class-map configuration mode.
Step 6	exit Example: <pre>Device(config)# exit</pre>	Returns to global configuration mode.
Step 7	class-map [match-all match-any] class-map-name Example: <pre>Device(config)# class-map match-all cm-2</pre>	<p>Creates an interface-level class map, and enters class-map configuration mode.</p> <p>By default, no class maps are defined.</p> <ul style="list-style-type: none"> (Optional) Use the match-all keyword to perform a logical-AND of all matching statements under this class map. All match criteria in the class map must be matched. (Optional) Use the match-any keyword to perform a logical-OR of all matching statements under this class map. One or more match criteria must be matched. For <i>class-map-name</i>, specify the name of the class map. <p>If neither the match-all or match-any keyword is specified, the default is match-all.</p> <p>Because only one match command per class map is supported, the match-all and match-any keywords function the same.</p>
Step 8	match input-interface interface-id-list Example: <pre>Device(config-cmap)# match input-interface gigabitethernet 3/0/1-3/0/2</pre>	<p>Specifies the physical ports on which the interface-level class map acts. You can specify up to six ports as follows:</p> <ul style="list-style-type: none"> A single port (counts as one entry) A list of ports separated by a space (each port counts as an entry)

	Command or Action	Purpose
		<ul style="list-style-type: none"> A range of ports separated by a hyphen (counts as two entries) <p>This command can only be used in the child-level policy map and must be the only match condition in the child-level policy map.</p>
Step 9	<p>exit</p> <p>Example:</p> <pre>Device(config-cmap)# exit</pre>	Returns to class-map configuration mode.
Step 10	<p>exit</p> <p>Example:</p> <pre>Device(config)# exit</pre>	Returns to global configuration mode.
Step 11	<p>policy-map <i>policy-map-name</i></p> <p>Example:</p> <pre>Device# policy-map port-plcmap</pre>	<p>Creates an interface-level policy map by entering the policy-map name, and enters policy-map configuration mode.</p> <p>By default, no policy maps are defined, and no policing is performed.</p> <p>To delete an existing policy map, use the no policy-map <i>policy-map-name</i> global configuration command. To delete an existing class map, use the no class <i>class-map-name</i> policy-map configuration command.</p> <p>To return to the untrusted state in a policy map, use the no trust policy-map configuration command. To remove an assigned DSCP or IP precedence value, use the no set {dscp new-dscp ip precedence new-precedence} policy-map configuration command.</p>
Step 12	<p>class-map <i>class-map-name</i></p> <p>Example:</p> <pre>Device(config-pmap)# class cm-interface-1</pre>	<p>Defines an interface-level traffic classification, and enters policy-map configuration mode.</p> <p>By default, no policy-map class-maps are defined.</p> <p>If a traffic class has already been defined by using the class-map global configuration command, specify its name for <i>class-map-name</i> in this command.</p>
Step 13	<p>police <i>rate-bps burst-byte</i> [exceed-action {drop policed-dscp-transmit}]</p> <p>Example:</p> <pre>Device(config-pmap-c)# police 900000 9000 exceed-action</pre>	<p>Defines an individual policer for the classified traffic.</p> <p>By default, no policer is defined.</p> <ul style="list-style-type: none"> For <i>rate-bps</i>, specify average traffic rate in bits per second (b/s). The range is 8000 to 10000000000.

	Command or Action	Purpose
	<code>policed-dscp-transmit</code>	<ul style="list-style-type: none"> For <i>burst-byte</i>, specify the normal burst size in bytes. The range is 8000 to 1000000. (Optional) Specifies the action to take when the rates are exceeded. Use the exceed-action drop keywords to drop the packet. Use the exceed-action policed-dscp-transmit keywords to mark down the DSCP value (by using the policed-DSCP map) and to send the packet. <p>To remove an existing policer in an interface-level policy map, use the no police rate-bps burst-byte [exceed-action {drop policed-dscp-transmit}] policy-map configuration command. To remove the hierarchical policy map and port associations, use the no service-policy input policy-map-name interface configuration command.</p>
Step 14	exit Example: <pre>Device(config-pmap-c)# exit</pre>	Returns to policy-map configuration mode.
Step 15	exit Example: <pre>Device(config-pmap)# exit</pre>	Returns to global configuration mode.
Step 16	policy-map <i>policy-map-name</i> Example: <pre>Device(config)# policy-map vlan-plcmap</pre>	<p>Creates a VLAN-level policy map by entering the policy-map name, and enters policy-map configuration mode.</p> <p>By default, no policy maps are defined.</p> <p>The default behavior of a policy map is to set the DSCP to 0 if the packet is an IP packet and to set the CoS to 0 if the packet is tagged. No policing is performed.</p>
Step 17	class [<i>class-map-name</i> class-default] Example: <pre>Device(config-pmap)# class cm-1</pre>	<p>Defines a VLAN-level traffic classification, and enters policy-map class configuration mode.</p> <p>By default, no policy-map class-maps are defined.</p> <p>If a traffic class has already been defined by using the class-map global configuration command, specify its name for class-map-name in this command.</p> <p>A class-default traffic class is pre-defined and can be added to any policy. It is always placed at the end of a policy map. With an implied match any included in the class-default class, all packets that have not already matched the other traffic classes will match class-default.</p>

	Command or Action	Purpose
Step 18	<p>trust [cos dscp ip-precedence]</p> <p>Example:</p> <pre>Device(config-pmap-c)# trust dscp</pre>	<p>Configures the trust state, which QoS uses to generate a CoS-based or DSCP-based QoS label.</p> <p>This command is mutually exclusive with the set command within the same policy map. If you enter the trust command, omit Step 18.</p> <p>By default, the port is not trusted. If no keyword is specified when the command is entered, the default is dscp.</p> <p>The keywords have these meanings:</p> <ul style="list-style-type: none"> • cos—QoS derives the DSCP value by using the received or default port CoS value and the CoS-to-DSCP map. • dscp—QoS derives the DSCP value by using the DSCP value from the ingress packet. For non-IP packets that are tagged, QoS derives the DSCP value by using the received CoS value; for non-IP packets that are untagged, QoS derives the DSCP value by using the default port CoS value. In either case, the DSCP value is derived from the CoS-to-DSCP map. • ip-precedence—QoS derives the DSCP value by using the IP precedence value from the ingress packet and the IP-precedence-to-DSCP map. For non-IP packets that are tagged, QoS derives the DSCP value by using the received CoS value; for non-IP packets that are untagged, QoS derives the DSCP value by using the default port CoS value. In either case, the DSCP value is derived from the CoS-to-DSCP map.
Step 19	<p>set {dscp <i>new-dscp</i> ip precedence <i>new-precedence</i>}</p> <p>Example:</p> <pre>Device(config-pmap-c)# set dscp 10</pre>	<p>Classifies IP traffic by setting a new value in the packet.</p> <ul style="list-style-type: none"> • For dscp <i>new-dscp</i>, enter a new DSCP value to be assigned to the classified traffic. The range is 0 to 63. • For ip precedence <i>new-precedence</i>, enter a new IP-precedence value to be assigned to the classified traffic. The range is 0 to 7.
Step 20	<p>service-policy <i>policy-map-name</i></p> <p>Example:</p> <pre>Device(config-pmap-c)# service-policy port-plcmap-1</pre>	<p>Specifies the interface-level policy-map name (from Step 10) and associate it with the VLAN-level policy map.</p> <p>If the VLAN-level policy map specifies more than one class, each class can have a different service-policy <i>policy-map-name</i> command.</p>
Step 21	<p>exit</p> <p>Example:</p>	<p>Returns to policy-map configuration mode.</p>

	Command or Action	Purpose
	Device (config-pmap-c) # exit	
Step 22	exit Example: Device (config-pmap) # exit	Returns to global configuration mode.
Step 23	interface <i>interface-id</i> Example: Device (config) # interface vlan 10	Specifies the SVI to which to attach the hierarchical policy map, and enters interface configuration mode.
Step 24	service-policy input <i>policy-map-name</i> Example: Device (config-if) # service-policy input vlan-plcmap	<p>Specifies the VLAN-level policy-map name, and applies it to the SVI. Repeat the previous step and this command to apply the policy map to other SVIs.</p> <p>If the hierarchical VLAN-level policy map has more than one interface-level policy map, all class maps must be configured to the same VLAN-level policy map specified in the service-policy <i>policy-map-name</i> command.</p> <p>Note To remove the hierarchical policy map and port associations, use the no service-policy input <i>policy-map-name</i> interface configuration command.</p>
Step 25	end Example: Device (config-if) # end	Returns to privileged EXEC mode.
Step 26	show policy-map [<i>policy-map-name</i> [class <i>class-map-name</i>]] or show mls qos vlan-based Example: Device# show mls qos vlan-based	Verifies your entries.
Step 27	copy running-config startup-config Example: Device# copy-running-config startup-config	(Optional) Saves your entries in the configuration file.

Classifying, Policing, and Marking Traffic by Using Aggregate Policers

By using an aggregate policer, you can create a policer that is shared by multiple traffic classes within the same policy map. However, you cannot use the aggregate policer across different policy maps or ports.

You can configure aggregate policers only in nonhierarchical policy maps on physical ports.

SUMMARY STEPS

1. **configure terminal**
2. **mls qos aggregate-policer** *aggregate-policer-name* *rate-bps* *burst-byte* **exceed-action** {**drop** | **policed-dscp-transmit**}
3. **class-map** [**match-all** | **match-any**] *class-map-name*
4. **policy-map** *policy-map-name*
5. **class** [*class-map-name* | **class-default**]
6. **police aggregate** *aggregate-policer-name*
7. **exit**
8. **interface** *interface-id*
9. **service-policy input** *policy-map-name*
10. **end**
11. **show mls qos aggregate-policer** [*aggregate-policer-name*]
12. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p>configure terminal</p> <p>Example:</p> <pre>Device# configure terminal</pre>	Enters global configuration mode.
Step 2	<p>mls qos aggregate-policer <i>aggregate-policer-name</i> <i>rate-bps</i> <i>burst-byte</i> exceed-action {drop policed-dscp-transmit}</p> <p>Example:</p> <pre>Device(config)# mls qos aggregate-police transmit1 48000 8000 exceed-action policed-dscp-transmit</pre>	<p>Defines the policer parameters that can be applied to multiple traffic classes within the same policy map.</p> <p>By default, no aggregate policer is defined.</p> <ul style="list-style-type: none"> • For <i>aggregate-policer-name</i>, specify the name of the aggregate policer. • For <i>rate-bps</i>, specify average traffic rate in bits per second (b/s). The range is 8000 to 10000000000. • For <i>burst-byte</i>, specify the normal burst size in bytes. The range is 8000 to 1000000. • Specifies the action to take when the rates are exceeded. Use the exceed-action drop keywords to drop the packet. Use the exceed-action policed-dscp-transmit keywords to mark down the DSCP value (by using the policed-DSCP map) and to send the packet.

	Command or Action	Purpose
Step 3	class-map [match-all match-any] <i>class-map-name</i> Example: Device(config)# class-map ipclass1	Creates a class map to classify traffic as necessary.
Step 4	policy-map <i>policy-map-name</i> Example: Device(config-cmap)# policy-map aggflow1	Creates a policy map by entering the policy map name, and enters policy-map configuration mode.
Step 5	class [<i>class-map-name</i> class-default] Example: Device(config-cmap-p)# class ipclass1	Defines a traffic classification, and enters policy-map class configuration mode.
Step 6	police aggregate <i>aggregate-policer-name</i> Example: Device(configure-cmap-p)# police aggregate transmit1	Applies an aggregate policer to multiple classes in the same policy map. For <i>aggregate-policer-name</i> , enter the name specified in Step 2. To remove the specified aggregate policer from a policy map, use the no police aggregate <i>aggregate-policer-name</i> policy map configuration command. To delete an aggregate policer and its parameters, use the no mls qos aggregate-policer <i>aggregate-policer-name</i> global configuration command.
Step 7	exit Example: Device(configure-cmap-p)# exit	Returns to global configuration mode.
Step 8	interface <i>interface-id</i> Example: Device(config)# interface gigabitethernet 2/0/1	Specifies the port to attach to the policy map, and enters interface configuration mode. Valid interfaces include physical ports.
Step 9	service-policy input <i>policy-map-name</i> Example: Device(config-if)# service-policy input aggflow1	Specifies the policy-map name, and applies it to an ingress port. Only one policy map per ingress port is supported.

	Command or Action	Purpose
Step 10	end Example: <pre>Device(configure-if)# end</pre>	Returns to privileged EXEC mode.
Step 11	show mls qos aggregate-policer [<i>aggregate-policer-name</i>] Example: <pre>Device# show mls qos aggregate-policer transmit1</pre>	Verifies your entries.
Step 12	copy running-config startup-config Example: <pre>Device# copy-running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Configuring DSCP Maps

Configuring the CoS-to-DSCP Map

You use the CoS-to-DSCP map to map CoS values in incoming packets to a DSCP value that QoS uses internally to represent the priority of the traffic.

Beginning in privileged EXEC mode, follow these steps to modify the CoS-to-DSCP map. This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **mls qos map cos-dscp** *dscp1...dscp8*
3. **end**
4. **show mls qos maps cos-dscp**
5. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>Device# configure terminal</pre>	Enters global configuration mode.
Step 2	mls qos map cos-dscp <i>dscp1...dscp8</i> Example:	Modifies the CoS-to-DSCP map.

	Command or Action	Purpose
	<pre>Device(config)# mls qos map cos-dscp 10 15 20 25 30 35 40 45</pre>	<p>For <i>dscp1...dscp8</i>, enter eight DSCP values that correspond to CoS values 0 to 7. Separate each DSCP value with a space.</p> <p>The DSCP range is 0 to 63.</p> <p>Note To return to the default map, use the no mls qos cos-dscp global configuration command.</p>
Step 3	<p>end</p> <p>Example:</p> <pre>Device(config)# end</pre>	Returns to privileged EXEC mode.
Step 4	<p>show mls qos maps cos-dscp</p> <p>Example:</p> <pre>Device# show mls qos maps cos-dscp</pre>	Verifies your entries.
Step 5	<p>copy running-config startup-config</p> <p>Example:</p> <pre>Device# copy-running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Configuring the IP-Precedence-to-DSCP Map

You use the IP-precedence-to-DSCP map to map IP precedence values in incoming packets to a DSCP value that QoS uses internally to represent the priority of the traffic.

Beginning in privileged EXEC mode, follow these steps to modify the IP-precedence-to-DSCP map. This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **mls qos map ip-prec-dscp *dscp1...dscp8***
3. **end**
4. **show mls qos maps ip-prec-dscp**
5. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p>configure terminal</p> <p>Example:</p>	Enters global configuration mode.

	Command or Action	Purpose
	Device# <code>configure terminal</code>	
Step 2	mls qos map ip-prec-dscp <i>dscp1...dscp8</i> Example: <pre>Device(config)# mls qos map ip-prec-dscp 10 15 20 25 30 35 40 45</pre>	Modifies the IP-precedence-to-DSCP map. For <i>dscp1...dscp8</i> , enter eight DSCP values that correspond to the IP precedence values 0 to 7. Separate each DSCP value with a space. The DSCP range is 0 to 63. Note To return to the default map, use the no mls qos ip-prec-dscp global configuration command.
Step 3	end Example: <pre>Device(config)# end</pre>	Returns to privileged EXEC mode.
Step 4	show mls qos maps ip-prec-dscp Example: <pre>Device# show mls qos maps ip-prec-dscp</pre>	Verifies your entries.
Step 5	copy running-config startup-config Example: <pre>Device# copy-running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Configuring the Policed-DSCP Map

You use the policed-DSCP map to mark down a DSCP value to a new value as the result of a policing and marking action.

The default policed-DSCP map is a null map, which maps an incoming DSCP value to the same DSCP value.

Beginning in privileged EXEC mode, follow these steps to modify the policed-DSCP map. This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **mls qos map policed-dscp** *dscp-list* **to** *mark-down-dscp*
3. **end**
4. **show mls qos maps policed-dscp**
5. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode.
Step 2	mls qos map policed-dscp <i>dscp-list</i> to <i>mark-down-dscp</i> Example: Device(config)# <code>mls qos map policed-dscp 50 51 52 53 54 55 56 57 to 0</code>	Modifies the policed-DSCP map. <ul style="list-style-type: none"> • For <i>dscp-list</i>, enter up to eight DSCP values separated by spaces. Then enter the to keyword. • For <i>mark-down-dscp</i>, enter the corresponding policed (marked down) DSCP value. Note To return to the default map, use the no mls qos policed-dscp global configuration command.
Step 3	end Example: Device(config)# <code>end</code>	Returns to privileged EXEC mode.
Step 4	show mls qos maps policed-dscp Example: Device(config)# <code>show mls qos maps policed-dscp</code>	Verifies your entries.
Step 5	copy running-config startup-config Example: Device#	(Optional) Saves your entries in the configuration file.

Configuring the DSCP-to-CoS Map

You use the DSCP-to-CoS map to generate a CoS value, which is used to select one of the four egress queues.

Beginning in privileged EXEC mode, follow these steps to modify the DSCP-to-CoS map. This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **mls qos map dscp-cos *dscp-list* to *cos***
3. **end**

4. `show mls qos maps dscp-to-cos`
5. `copy running-config startup-config`

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p>configure terminal</p> <p>Example:</p> <pre>Device# configure terminal</pre>	Enters global configuration mode.
Step 2	<p>mls qos map dscp-cos <i>dscp-list</i> to <i>cos</i></p> <p>Example:</p> <pre>Device# mls qos map dscp-cos 0 8 16 24 32 40 48 50 to 0</pre>	<p>Modifies the DSCP-to-CoS map.</p> <ul style="list-style-type: none"> • For <i>dscp-list</i>, enter up to eight DSCP values separated by spaces. Then enter the to keyword. • For <i>cos</i>, enter the CoS value to which the DSCP values correspond. <p>The DSCP range is 0 to 63; the CoS range is 0 to 7.</p> <p>Note To return to the default map, use the no mls qos dscp-cos global configuration command.</p>
Step 3	<p>end</p> <p>Example:</p> <pre>Device(config)# end</pre>	Returns to privileged EXEC mode.
Step 4	<p>show mls qos maps dscp-to-cos</p> <p>Example:</p> <pre>Device# show mls qos maps dscp-to-cos</pre>	Verifies your entries.
Step 5	<p>copy running-config startup-config</p> <p>Example:</p> <pre>Device# copy-running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Configuring the DSCP-to-DSCP-Mutation Map

If two QoS domains have different DSCP definitions, use the DSCP-to-DSCP-mutation map to translate one set of DSCP values to match the definition of another domain. You apply the DSCP-to-DSCP-mutation map to the receiving port (ingress mutation) at the boundary of a QoS administrative domain.

With ingress mutation, the new DSCP value overwrites the one in the packet, and QoS applies the new value to the packet. The sends the packet out the port with the new DSCP value.

You can configure multiple DSCP-to-DSCP-mutation maps on an ingress port. The default DSCP-to-DSCP-mutation map is a null map, which maps an incoming DSCP value to the same DSCP value.

Beginning in privileged EXEC mode, follow these steps to modify the DSCP-to-DSCP-mutation map. This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **mls qos map dscp-mutation** *dscp-mutation-name in-dscp to out-dscp*
3. **interface** *interface-id*
4. **mls qos trust dscp**
5. **mls qos dscp-mutation** *dscp-mutation-name*
6. **end**
7. **show mls qos maps dscp-mutation**
8. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 2	mls qos map dscp-mutation <i>dscp-mutation-name in-dscp to out-dscp</i> Example: Device(config)# mls qos map dscp-mutation mutation1 1 2 3 4 5 6 7 to 0	Modifies the DSCP-to-DSCP-mutation map. <ul style="list-style-type: none"> • For <i>dscp-mutation-name</i>, enter the mutation map name. You can create more than one map by specifying a new name. • For <i>in-dscp</i>, enter up to eight DSCP values separated by spaces. Then enter the to keyword. • For <i>out-dscp</i>, enter a single DSCP value. The DSCP range is 0 to 63. Note To return to the default map, use the no mls qos dscp-mutation <i>dscp-mutation-name</i> global configuration command.
Step 3	interface <i>interface-id</i> Example: Device(config)# interface gigabitethernet1/0/1	Specifies the port to which to attach the map, and enters interface configuration mode. Valid interfaces include physical ports.

	Command or Action	Purpose
Step 4	mls qos trust dscp Example: <pre>Device(config-if)# mls qos trust dscp</pre>	Configures the ingress port as a DSCP-trusted port. By default, the port is not trusted.
Step 5	mls qos dscp-mutation <i>dscp-mutation-name</i> Example: <pre>Device(config-if)# mls qos dscp-mutation mutation1</pre>	Applies the map to the specified ingress DSCP-trusted port. For <i>dscp-mutation-name</i> , enter the mutation map name specified in Step 2.
Step 6	end Example: <pre>Device(config-if)# end</pre>	Returns to privileged EXEC mode.
Step 7	show mls qos maps dscp-mutation Example: <pre>Device# show mls qos maps dscp-mutation</pre>	Verifies your entries.
Step 8	copy running-config startup-config Example: <pre>Device# copy-running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Configuring Ingress Queue Characteristics

Depending on the complexity of your network and your QoS solution, you might need to perform all of the tasks in the next modules. You need to make decisions about these characteristics:

- Which packets are assigned (by DSCP or CoS value) to each queue?
- What drop percentage thresholds apply to each queue, and which CoS or DSCP values map to each threshold?
- How much of the available buffer space is allocated between the queues?
- How much of the available bandwidth is allocated between the queues?
- Is there traffic (such as voice) that should be given high priority?

Configuration Guidelines

Follow these guidelines when the expedite queue is enabled or the egress queues are serviced based on their SRR weights:

- If the egress expedite queue is enabled, it overrides the SRR shaped and shared weights for queue 1.
- If the egress expedite queue is disabled and the SRR shaped and shared weights are configured, the shaped mode overrides the shared mode for queue 1, and SRR services this queue in shaped mode.
- If the egress expedite queue is disabled and the SRR shaped weights are not configured, SRR services this queue in shared mode.

Mapping DSCP or CoS Values to an Ingress Queue and Setting WTD Thresholds

You can prioritize traffic by placing packets with particular DSCPs or CoSs into certain queues and adjusting the queue thresholds so that packets with lower priorities are dropped.

Beginning in privileged EXEC mode, follow these steps to map DSCP or CoS values to an ingress queue and to set WTD thresholds. This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. Use one of the following:
 - **mls qos srr-queue input dscp-map queue *queue-id* threshold *threshold-id* dscp1...dscp8**
 - **mls qos srr-queue input cos-map queue *queue-id* threshold *threshold-id* cos1...cos8**
3. **mls qos srr-queue input threshold *queue-id* threshold-percentage1 threshold-percentage2**
4. **end**
5. **show mls qos maps**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 2	Use one of the following: <ul style="list-style-type: none"> • mls qos srr-queue input dscp-map queue <i>queue-id</i> threshold <i>threshold-id</i> dscp1...dscp8 • mls qos srr-queue input cos-map queue <i>queue-id</i> threshold <i>threshold-id</i> cos1...cos8 Example: Device(config)# mls qos srr-queue input	Maps DSCP or CoS values to an ingress queue and to a threshold ID. By default, DSCP values 0–39 and 48–63 are mapped to queue 1 and threshold 1. DSCP values 40–47 are mapped to queue 2 and threshold 1. By default, CoS values 0–4, 6, and 7 are mapped to queue 1 and threshold 1. CoS value 5 is mapped to queue 2 and threshold 1.

	Command or Action	Purpose
	<code>dscp-map queue 1 threshold 2 20 21 22 23 24 25 26</code>	<ul style="list-style-type: none"> For <i>queue-id</i>, the range is 1 to 2. For <i>threshold-id</i>, the range is 1 to 3. The drop-threshold percentage for threshold 3 is predefined. It is set to the queue-full state. For <i>dscp1...dscp8</i>, enter up to eight values, and separate each value with a space. The range is 0 to 63. For <i>cos1...cos8</i>, enter up to eight values, and separate each value with a space. The range is 0 to 7.
Step 3	<p>mls qos srr-queue input threshold <i>queue-id</i> <i>threshold-percentage1</i> <i>threshold-percentage2</i></p> <p>Example:</p> <pre>Device(config)# mls qos srr-queue input threshold 1 50 70</pre>	<p>Assigns the two WTD threshold percentages for (threshold 1 and 2) to an ingress queue. The default, both thresholds are set to 100 percent.</p> <ul style="list-style-type: none"> For <i>queue-id</i>, the range is 1 to 2. For <i>threshold-percentage1</i> <i>threshold-percentage2</i>, the range is 1 to 100. Separate each value with a space. <p>Each threshold value is a percentage of the total number of queue descriptors allocated for the queue.</p>
Step 4	<p>end</p> <p>Example:</p> <pre>Device(config)# end</pre>	Returns to privileged EXEC mode.
Step 5	<p>show mls qos maps</p> <p>Example:</p> <pre>Device# show mls qos maps</pre>	<p>Verifies your entries.</p> <p>The DSCP input queue threshold map appears as a matrix. The d1 column specifies the most-significant digit of the DSCP number; the d2 row specifies the least-significant digit in the DSCP number. The intersection of the d1 and the d2 values provides the queue ID and threshold ID; for example, queue 2 and threshold 1 (02-01).</p> <p>The CoS input queue threshold map shows the CoS value in the top row and the corresponding queue ID and threshold ID in the second row; for example, queue 2 and threshold 2 (2-2).</p>
Step 6	<p>copy running-config startup-config</p> <p>Example:</p> <pre>Device# copy running-config startup-config</pre>	<p>(Optional) Saves your entries in the configuration file.</p> <p>To return to the default CoS input queue threshold map or the default DSCP input queue threshold map, use the no mls qos srr-queue input cos-map or the no mls qos srr-queue input dscp-map global configuration command. To return to the default WTD threshold percentages, use the no mls qos srr-queue input threshold queue-id global configuration command</p>

Allocating Buffer Space Between the Ingress Queues

You define the ratio (allocate the amount of space) with which to divide the ingress buffers between the two queues. The buffer and the bandwidth allocation control how much data can be buffered before packets are dropped.

Beginning in privileged EXEC mode, follow these steps to allocate the buffers between the ingress queues. This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **mls qos srr-queue input buffers *percentage1 percentage2***
3. **end**
4. Use one of the following:
 - **show mls qos interface buffer**
 - **show mls qos input-queue**
5. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 2	mls qos srr-queue input buffers <i>percentage1 percentage2</i> Example: Device(config)# mls qos srr-queue input buffers 60 40	Allocates the buffers between the ingress queues By default 90 percent of the buffers are allocated to queue 1, and 10 percent of the buffers are allocated to queue 2. For <i>percentage1 percentage2</i> , the range is 0 to 100. Separate each value with a space. You should allocate the buffers so that the queues can handle any incoming bursty traffic.
Step 3	end Example: Device(config)# end	Returns to privileged EXEC mode.
Step 4	Use one of the following: <ul style="list-style-type: none"> • show mls qos interface buffer • show mls qos input-queue Example:	Verifies your entries.

	Command or Action	Purpose
	Device# <code>show mls qos interface buffer</code> or Device# <code>show mls qos input-queue</code>	
Step 5	copy running-config startup-config Example: Device# <code>copy-running-config startup-config</code>	(Optional) Saves your entries in the configuration file. To return to the default setting, use the no mls qos srr-queue input buffers global configuration command.

Allocating Bandwidth Between the Ingress Queues

You need to specify how much of the available bandwidth is allocated between the ingress queues. The ratio of the weights is the ratio of the frequency in which the SRR scheduler sends packets from each queue. The bandwidth and the buffer allocation control how much data can be buffered before packets are dropped. On ingress queues, SRR operates only in shared mode.



Note SRR bandwidth limit works in both mls qos enabled and disabled states.

Beginning in privileged EXEC mode, follow these steps to allocate bandwidth between the ingress queues. This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **mls qos srr-queue input bandwidth** *weight1 weight2*
3. **end**
4. Use one of the following:
 - **show mls qos interface queueing**
 - **show mls qos input-queue**
5. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode.
Step 2	mls qos srr-queue input bandwidth <i>weight1 weight2</i>	Assigns shared round robin weights to the ingress queues.

	Command or Action	Purpose
	<p>Example:</p> <pre>Device(config)# mls qos srr-queue input bandwidth 25 75</pre>	<p>The default setting for <i>weight1</i> and <i>weight2</i> is 4 (1/2 of the bandwidth is equally shared between the two queues).</p> <p>For <i>weight1</i> and <i>weight2</i>, the range is 1 to 100. Separate each value with a space.</p> <p>SRR services the priority queue for its configured weight as specified by the bandwidth keyword in the mls qos srr-queue input priority-queue queue-id bandwidth weight global configuration command. Then, SRR shares the remaining bandwidth with both ingress queues and services them as specified by the weights configured with the mls qos srr-queue input bandwidth weight1 weight2 global configuration command.</p>
Step 3	<p>end</p> <p>Example:</p> <pre>Device(config)# end</pre>	Returns to privileged EXEC mode.
Step 4	<p>Use one of the following:</p> <ul style="list-style-type: none"> • show mls qos interface queueing • show mls qos input-queue <p>Example:</p> <pre>Device# show mls qos interface queueing</pre> <p>OR</p> <pre>Device# show mls qos input-queue</pre>	Verifies your entries.
Step 5	<p>copy running-config startup-config</p> <p>Example:</p> <pre>Device# copy running-config startup-config</pre>	<p>(Optional) Saves your entries in the configuration file.</p> <p>To return to the default setting, use the no mls qos srr-queue input bandwidth global configuration command.</p>

Configuring Egress Queue Characteristics

Depending on the complexity of your network and your QoS solution, you might need to perform all of the tasks in the following modules. You need to make decisions about these characteristics:

- Which packets are mapped by DSCP or CoS value to each queue and threshold ID?
- What drop percentage thresholds apply to the queue-set (four egress queues per port), and how much reserved and maximum memory is needed for the traffic type?
- How much of the fixed buffer space is allocated to the queue-set?

- Does the bandwidth of the port need to be rate limited?
- How often should the egress queues be serviced and which technique (shaped, shared, or both) should be used?

Configuration Guidelines

Follow these guidelines when the expedite queue is enabled or the egress queues are serviced based on their SRR weights:

- If the egress expedite queue is enabled, it overrides the SRR shaped and shared weights for queue 1.
- If the egress expedite queue is disabled and the SRR shaped and shared weights are configured, the shaped mode overrides the shared mode for queue 1, and SRR services this queue in shaped mode.
- If the egress expedite queue is disabled and the SRR shaped weights are not configured, SRR services this queue in shared mode.

Allocating Buffer Space to and Setting WTD Thresholds for an Egress Queue Set

You can guarantee the availability of buffers, set WTD thresholds, and configure the maximum allocation for a queue set by using the **mls qos queue-set output *qset-id* threshold *queue-id* drop-threshold1 drop-threshold2 reserved-threshold maximum-threshold** global configuration command.

Each threshold value is a percentage of the queue's allocated buffers, which you specify by using the **mls qos queue-set output *qset-id* buffers *allocation1* ... *allocation4*** global configuration command. The queues use WTD to support distinct drop percentages for different traffic classes.



Note The egress queue default settings are suitable for most situations. You should change them only when you have a thorough understanding of the egress queues and if these settings do not meet your QoS solution.

Beginning in privileged EXEC mode, follow these steps to configure the memory allocation and to drop thresholds for a queue set. This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **mls qos srr-queue output queues 8**
3. **mls qos queue-set output *qset-id* buffers *allocation1* ... *allocation8***
4. **mls qos queue-set output *qset-id* threshold *queue-id* drop-threshold1 drop-threshold2 reserved-threshold maximum-threshold**
5. **interface *interface-id***
6. **queue-set *qset-id***
7. **end**
8. **show mls qos interface [*interface-id*] buffers**
9. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode.
Step 2	mls qos srr-queue output queues 8 Example: Device(config)# <code>mls qos srr-queue output queues 8</code>	<p>(Optional) The switch supports 4 egress queues by default, although you can enable a total of 8 egress queues. Use the optional mls qos srr-queue output queues 8 command to enable the additional 4 egress queues.</p> <p>Once 8 queue support is enabled, you can then proceed to configure the additional 4 queues. Any existing egress queue configuration commands are then modified to support the additional queue parameters.</p> <p>Note The option to enable 8 queues is only available on a standalone switch. If the switch is within a stack, then only an additional 2 queues can be enabled for a total of 6 egress queues.</p>
Step 3	mls qos queue-set output <i>qset-id</i> buffers <i>allocation1</i> ... <i>allocation8</i> Example: Device(config)# <code>mls qos queue-set output 2 buffers 40 20 20 20 10 10 10 10</code>	<p>Allocates buffers to a queue set.</p> <p>By default, all allocation values are equally mapped among the four queues (25, 25, 25, 25). Each queue has 1/4 of the buffer space. When eight egress queues are configured, then by default 30 percent of the total buffer space is allocated to queue 2 and 10 percent (each) to queues 1,3,4,5,6,7, and 8.</p> <p>If you enabled 8 egress queues as described in Step 2 above, then the following applies:</p> <ul style="list-style-type: none"> • For <i>qset-id</i>, enter the ID of the queue set. The range is 1 to 2. Each port belongs to a queue set, which defines all the characteristics of the four egress queues per port. • For <i>allocation1</i> ... <i>allocation8</i>, specify eight percentages, one for each queue in the queue set. For <i>allocation1</i>, <i>allocation3</i>, and <i>allocation4</i> to <i>allocation8</i>, the range is 0 to 99. For <i>allocation2</i>, the range is 1 to 100 (including the CPU buffer). <p>Allocate buffers according to the importance of the traffic; for example, give a large percentage of the buffer to the queue with the highest-priority traffic.</p> <p>Note To return to the default setting, use the no mls qos queue-set output <i>qset-id</i> buffers global configuration command.</p>

	Command or Action	Purpose
Step 4	<p>mls qos queue-set output <i>qset-id</i> threshold <i>queue-id</i> <i>drop-threshold1</i> <i>drop-threshold2</i> <i>reserved-threshold</i> <i>maximum-threshold</i></p> <p>Example:</p> <pre>Device(config)# mls qos queue-set output 2 threshold 2 40 60 80 200</pre>	<p>Configures the WTD thresholds, guarantee the availability of buffers, and configures the maximum memory allocation for the queue set (four egress queues per port).</p> <p>By default, the WTD thresholds for queues 1, 3, and 4 are set to 100 percent. The thresholds for queue 2 are set to 200 percent. The reserved thresholds for queues 1, 2, 3, and 4 are set to 50 percent. The maximum thresholds for all queues are set to 400 percent by default.</p> <p>If you enabled 8 egress queues as described in Step 2 above, then the following applies:</p> <ul style="list-style-type: none"> • For <i>qset-id</i>, enter the ID of the queue set specified in Step 3. The range is 1 to 2. • For <i>queue-id</i>, enter the specific queue in the queue set on which the command is performed. The <i>queue-id</i> range is 1-4 by default and 1-8 when 8 queues are enabled. • For <i>drop-threshold1</i> <i>drop-threshold2</i>, specify the two WTD thresholds expressed as a percentage of the queue's allocated memory. The range is 1 to 3200 percent. • For <i>reserved-threshold</i>, enter the amount of memory to be guaranteed (reserved) for the queue expressed as a percentage of the allocated memory. The range is 1 to 100 percent. • For <i>maximum-threshold</i>, enable a queue in the full condition to obtain more buffers than are reserved for it. This is the maximum memory the queue can have before the packets are dropped if the common pool is not empty. The range is 1 to 3200 percent. <p>Note To return to the default WTD threshold percentages, use the no mls qos queue-set output <i>qset-id</i> threshold [<i>queue-id</i>] global configuration command.</p>
Step 5	<p>interface <i>interface-id</i></p> <p>Example:</p> <pre>Device(config)# interface gigabitethernet1/0/1</pre>	<p>Specifies the port of the outbound traffic, and enters interface configuration mode.</p>
Step 6	<p>queue-set <i>qset-id</i></p> <p>Example:</p>	<p>Maps the port to a queue set.</p> <p>For <i>qset-id</i>, enter the ID of the queue set specified in Step 2. The range is 1 to 2. The default is 1.</p>

	Command or Action	Purpose
	Device(config-id)# queue-set 2	
Step 7	end Example: Device(config-id)# end	Returns to privileged EXEC mode.
Step 8	show mls qos interface [interface-id] buffers Example: Device# show mls qos interface buffers	Verifies your entries.
Step 9	copy running-config startup-config Example: Device# copy-running-config startup-config	(Optional) Saves your entries in the configuration file. To return to the default setting, use the no mls qos queue-set output qset-id buffers global configuration command. To return to the default WTD threshold percentages, use the no mls qos queue-set output qset-id threshold [queue-id] global configuration command.

Mapping DSCP or CoS Values to an Egress Queue and to a Threshold ID

You can prioritize traffic by placing packets with particular DSCPs or costs of service into certain queues and adjusting the queue thresholds so that packets with lower priorities are dropped.



Note The egress queue default settings are suitable for most situations. You should change them only when you have a thorough understanding of egress queues and if these settings do not meet your QoS solution.

Beginning in privileged EXEC mode, follow these steps to map DSCP or CoS values to an egress queue and to a threshold ID. This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. Use one of the following:
 - **mls qos srr-queue output dscp-map queue queue-id threshold threshold-id dscp1...dscp8**
 - **mls qos srr-queue output cos-map queue queue-id threshold threshold-id cos1...cos8**
3. **mls qos srr-queue output cos-map queue queue-id threshold threshold-id cos1...cos8**
4. **end**
5. **show mls qos maps**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>Device# configure terminal</pre>	Enters global configuration mode.
Step 2	Use one of the following: <ul style="list-style-type: none"> • mls qos srr-queue output dscp-map queue <i>queue-id</i> threshold <i>threshold-id</i> dscp1...dscp8 • mls qos srr-queue output cos-map queue <i>queue-id</i> threshold <i>threshold-id</i> cos1...cos8 Example: <pre>Device(config)# mls qos srr-queue output dscp-map queue 1 threshold 2 10 11</pre>	Maps DSCP or CoS values to an egress queue and to a threshold ID. By default, DSCP values 0–15 are mapped to queue 2 and threshold 1. DSCP values 16–31 are mapped to queue 3 and threshold 1. DSCP values 32–39 and 48–63 are mapped to queue 4 and threshold 1. DSCP values 40–47 are mapped to queue 1 and threshold 1. By default, CoS values 0 and 1 are mapped to queue 2 and threshold 1. CoS values 2 and 3 are mapped to queue 3 and threshold 1. CoS values 4, 6, and 7 are mapped to queue 4 and threshold 1. CoS value 5 is mapped to queue 1 and threshold 1. <ul style="list-style-type: none"> • For <i>queue-id</i>, the range is 1 to 4. • For <i>threshold-id</i>, the range is 1 to 3. The drop-threshold percentage for threshold 3 is predefined. It is set to the queue-full state. • For <i>dscp1...dscp8</i>, enter up to eight values, and separate each value with a space. The range is 0 to 63. • For <i>cos1...cos8</i>, enter up to eight values, and separate each value with a space. The range is 0 to 7. Note To return to the default DSCP output queue threshold map or the default CoS output queue threshold map, use the no mls qos srr-queue output dscp-map or the no mls qos srr-queue output cos-map global configuration command.
Step 3	mls qos srr-queue output cos-map queue <i>queue-id</i> threshold <i>threshold-id</i> cos1...cos8 Example: <pre>Device(config)# mls qos srr-queue output cos-map queue 3 threshold 1 2 3</pre>	Maps CoS values to an egress queue and to a threshold ID. By default, CoS values 0 and 1 are mapped to queue 2 and threshold 1. CoS values 2 and 3 are mapped to queue 3 and threshold 1. CoS values 4, 6, and 7 are mapped to queue 4 and threshold 1. CoS value 5 is mapped to queue 1 and threshold 1. <ul style="list-style-type: none"> • For <i>queue-id</i>, the range is 1 to 4. • For <i>threshold-id</i>, the range is 1 to 3. The drop-threshold percentage for threshold 3 is predefined. It is set to the queue-full state.

	Command or Action	Purpose
		<ul style="list-style-type: none"> For <i>cos1...cos8</i>, enter up to eight values, and separate each value with a space. The range is 0 to 7. <p>Note To return to the default CoS output queue threshold map, use the no mls qos srr-queue output cos-map global configuration command.</p>
Step 4	end Example: <pre>Device(config)# end</pre>	Returns to privileged EXEC mode.
Step 5	show mls qos maps Example: <pre>Device# show mls qos maps</pre>	<p>Verifies your entries.</p> <p>The DSCP output queue threshold map appears as a matrix. The d1 column specifies the most-significant digit of the DSCP number; the d2 row specifies the least-significant digit in the DSCP number. The intersection of the d1 and the d2 values provides the queue ID and threshold ID; for example, queue 2 and threshold 1 (02-01).</p> <p>The CoS output queue threshold map shows the CoS value in the top row and the corresponding queue ID and threshold ID in the second row; for example, queue 2 and threshold 2 (2-2).</p>
Step 6	copy running-config startup-config Example: <pre>Device# copy-running-config startup-config</pre>	<p>(Optional) Saves your entries in the configuration file.</p> <p>To return to the default DSCP output queue threshold map or the default CoS output queue threshold map, use the no mls qos srr-queue output dscp-map or the no mls qos srr-queue output cos-map global configuration command.</p>

Configuring SRR Shaped Weights on Egress Queues

You can specify how much of the available bandwidth is allocated to each queue. The ratio of the weights is the ratio of frequency in which the SRR scheduler sends packets from each queue.

You can configure the egress queues for shaped or shared weights, or both. Use shaping to smooth bursty traffic or to provide a smoother output over time.

Beginning in privileged EXEC mode, follow these steps to assign the shaped weights and to enable bandwidth shaping on the four egress queues mapped to a port. This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **srr-queue bandwidth shape** *weight1 weight2 weight3 weight4*
4. **end**

5. `show mls qos interface interface-id queueing`
6. `copy running-config startup-config`

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode.
Step 2	interface <i>interface-id</i> Example: Device(config)# <code>interface gigabitethernet2/0/1</code>	Specifies the port of the outbound traffic, and enters interface configuration mode.
Step 3	srr-queue bandwidth shape <i>weight1 weight2 weight3 weight4</i> Example: Device(config-if)# <code>srr-queue bandwidth shape 8 0 0 0</code>	<p>Assigns SRR weights to the egress queues. By default, <i>weight1</i> is set to 25; <i>weight2</i>, <i>weight3</i>, and <i>weight4</i> are set to 0, and these queues are in shared mode.</p> <p>For <i>weight1 weight2 weight3 weight4</i>, enter the weights to control the percentage of the port that is shaped. The inverse ratio (1/weight) controls the shaping bandwidth for this queue. Separate each value with a space. The range is 0 to 65535.</p> <p>If you configure a weight of 0, the corresponding queue operates in shared mode. The weight specified with the srr-queue bandwidth shape command is ignored, and the weights specified with the srr-queue bandwidth share interface configuration command for a queue come into effect. When configuring queues in the same queue-set for both shaping and sharing, make sure that you configure the lowest number queue for shaping.</p> <p>The shaped mode overrides the shared mode.</p> <p>To return to the default setting, use the no srr-queue bandwidth shape interface configuration command.</p>
Step 4	end Example: Device(config-if)# <code>end</code>	Returns to privileged EXEC mode.
Step 5	show mls qos interface <i>interface-id</i> queueing Example:	Verifies your entries.

	Command or Action	Purpose
	Device# <code>show mls qos interface interface-id queuing</code>	
Step 6	copy running-config startup-config Example: Device# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file. To return to the default setting, use the no srr-queue bandwidth shape interface configuration command.

Configuring SRR Shared Weights on Egress Queues

In shared mode, the queues share the bandwidth among them according to the configured weights. The bandwidth is guaranteed at this level but not limited to it. For example, if a queue empties and does not require a share of the link, the remaining queues can expand into the unused bandwidth and share it among them. With sharing, the ratio of the weights controls the frequency of dequeuing; the absolute values are meaningless.



Note The egress queue default settings are suitable for most situations. You should change them only when you have a thorough understanding of the egress queues and if these settings do not meet your QoS solution.

Beginning in privileged EXEC mode, follow these steps to assign the shared weights and to enable bandwidth sharing on the four egress queues mapped to a port. This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **srr-queue bandwidth share** *weight1 weight2 weight3 weight4*
4. **end**
5. **show mls qos interface** *interface-id queuing*
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode.
Step 2	interface <i>interface-id</i> Example: Device(config)# <code>interface</code>	Specifies the port of the outbound traffic, and enters interface configuration mode.

	Command or Action	Purpose
	<code>gigabitethernet2/0/1</code>	
Step 3	<p>srr-queue bandwidth share <i>weight1 weight2 weight3 weight4</i></p> <p>Example:</p> <pre>Device(config-id)# srr-queue bandwidth share 1 2 3 4</pre>	<p>Assigns SRR weights to the egress queues. By default, all four weights are 25 (1/4 of the bandwidth is allocated to each queue).</p> <p>For <i>weight1 weight2 weight3 weight4</i>, enter the weights to control the ratio of the frequency in which the SRR scheduler sends packets. Separate each value with a space. The range is 1 to 255.</p> <p>To return to the default setting, use the no srr-queue bandwidth share interface configuration command.</p>
Step 4	<p>end</p> <p>Example:</p> <pre>Device(config-id)# end</pre>	Returns to privileged EXEC mode.
Step 5	<p>show mls qos interface <i>interface-id queuing</i></p> <p>Example:</p> <pre>Device# show mls qos interface interface_id queuing</pre>	Verifies your entries.
Step 6	<p>copy running-config startup-config</p> <p>Example:</p> <pre>Device# copy-running-config startup-config</pre>	<p>(Optional) Saves your entries in the configuration file.</p> <p>To return to the default setting, use the no srr-queue bandwidth share interface configuration command.</p>

Configuring the Egress Expedite Queue

You can ensure that certain packets have priority over all others by queuing them in the egress expedite queue. SRR services this queue until it is empty before servicing the other queues.

Beginning in privileged EXEC mode, follow these steps to enable the egress expedite queue. This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **mls qos**
3. **interface** *interface-id*
4. **priority-queue out**
5. **end**
6. **show running-config**

7. copy running-config startup-config

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 2	mls qos Example: Device(config)# mls qos	Enables QoS on a switch.
Step 3	interface interface-id Example: Device(config)# interface gigabitethernet1/0/1	Specifies the egress port, and enters interface configuration mode.
Step 4	priority-queue out Example: Device(config-if)# priority-queue out	<p>Enables the egress expedite queue, which is disabled by default.</p> <p>When you configure this command, the SRR weight and queue size ratios are affected because there is one fewer queue participating in SRR. This means that <i>weight1</i> in the srr-queue bandwidth shape or the srr-queue bandwidth share command is ignored (not used in the ratio calculation).</p> <p>Note To disable the egress expedite queue, use the no priority-queue out interface configuration command.</p>
Step 5	end Example: Device(config-if)# end	Returns to privileged EXEC mode.
Step 6	show running-config Example: Device# show running-config	Verifies your entries.
Step 7	copy running-config startup-config	(Optional) Saves your entries in the configuration file.

	Command or Action	Purpose
	Example: Device# <code>copy running-config startup-config</code>	To disable the egress expedite queue, use the no priority-queue out interface configuration command.

Limiting the Bandwidth on an Egress Interface

You can limit the bandwidth on an egress port. For example, if a customer pays only for a small percentage of a high-speed link, you can limit the bandwidth to that amount.



Note The egress queue default settings are suitable for most situations. You should change them only when you have a thorough understanding of the egress queues and if these settings do not meet your QoS solution.

Beginning in privileged EXEC mode, follow these steps to limit the bandwidth on an egress port. This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **srr-queue bandwidth limit** *weight1*
4. **end**
5. **show mls qos interface** [*interface-id*] **queueing**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode.
Step 2	interface <i>interface-id</i> Example: Device(config)# <code>interface gigabitethernet2/0/1</code>	Specifies the port to be rate-limited, and enters interface configuration mode.
Step 3	srr-queue bandwidth limit <i>weight1</i> Example: Device(config-if)# <code>srr-queue</code>	Specifies the percentage of the port speed to which the port should be limited. The range is 10 to 90. By default, the port is not rate-limited and is set to 100 percent.

	Command or Action	Purpose
	<code>bandwidth limit 80</code>	Note To return to the default setting, use the no srr-queue bandwidth limit interface configuration command.
Step 4	end Example: Device(config-if)# end	Returns to privileged EXEC mode.
Step 5	show mls qos interface [<i>interface-id</i>] queueing Example: Device# show mls qos interface interface_id queueing	Verifies your entries.
Step 6	copy running-config startup-config Example: Device# copy-running-config startup-config	(Optional) Saves your entries in the configuration file. To return to the default setting, use the no srr-queue bandwidth limit interface configuration command.

Monitoring Standard QoS

Table 86: Commands for Monitoring Standard QoS on the Switch

Command	Description
show class-map [<i>class-map-name</i>]	Displays QoS class maps, which define the match criteria to classify traffic.
show mls qos	Displays global QoS configuration information.
show mls qos aggregate-policer [<i>aggregate-policer-name</i>]	Displays the aggregate policer configuration.
show mls qos interface [<i>interface-id</i>] [buffers policers queueing statistics]	Displays QoS information at the port level, including the buffer allocation, which ports have configured policers, the queueing strategy, and the ingress and egress statistics.
show mls qos maps [cos-dscp cos-output-q dscp-cos dscp-mutation <i>dscp-mutation-name</i> dscp-output-q ip-prec-dscp policed-dscp]	Displays QoS mapping information.
show mls qos queue-set [<i>qset-id</i>]	Displays QoS settings for the egress queues.

Command	Description
<code>show mls qos vlan <i>vlan-id</i></code>	Displays the policy maps attached to the specified SVI.
<code>show policy-map [<i>policy-map-name</i> [class <i>class-map-name</i>]]</code>	Displays QoS policy maps, which define classification criteria for incoming traffic. Do not use the show policy-map interface privileged EXEC command to display classification information for incoming traffic. The control-plane and interface keywords are not supported, and the statistics shown in the display should be ignored.
<code>show running-config include rewrite</code>	Displays the DSCP transparency setting.

Configuration Examples for QoS

Example: Configuring Port to the DSCP-Trusted State and Modifying the DSCP-to-DSCP-Mutation Map

This example shows how to configure a port to the DSCP-trusted state and to modify the DSCP-to-DSCP-mutation map (named *gi1/0/2-mutation*) so that incoming DSCP values 10 to 13 are mapped to DSCP 30:

```
Device(config)# mls qos map dscp-mutation gigabitethernet1/0/2-mutation
10 11 12 13 to 30
Device(config)# interface gigabitethernet1/0/2
Device(config-if)# mls qos trust dscp
Device(config-if)# mls qos dscp-mutation gigabitethernet1/0/2-mutation
Device(config-if)# end
```

Examples: Classifying Traffic by Using ACLs

This example shows how to allow access for only those hosts on the three specified networks. The wildcard bits apply to the host portions of the network addresses. Any host with a source address that does not match the access list statements is rejected.

```
Device(config)# access-list 1 permit 192.5.255.0 0.0.0.255
Device(config)# access-list 1 permit 128.88.0.0 0.0.255.255
Device(config)# access-list 1 permit 36.0.0.0 0.0.0.255
! (Note: all other access implicitly denied)
```

This example shows how to create an ACL that permits IP traffic from any source to any destination that has the DSCP value set to 32:

```
Device(config)# access-list 100 permit ip any any dscp 32
```

This example shows how to create an ACL that permits IP traffic from a source host at 10.1.1.1 to a destination host at 10.1.1.2 with a precedence value of 5:

```
Device(config)# access-list 100 permit ip host 10.1.1.1 host 10.1.1.2 precedence 5
```

This example shows how to create an ACL that permits PIM traffic from any source to a destination group address of 224.0.0.2 with a DSCP set to 32:

```
Device(config)# access-list 102 permit pim any 224.0.0.2 dscp 32
```

This example shows how to create an ACL that permits IPv6 traffic from any source to any destination that has the DSCP value set to 32:

```
Device(config)# ipv6 access-list 100 permit ip any any dscp 32
```

This example shows how to create an ACL that permits IPv6 traffic from a source host at 10.1.1.1 to a destination host at 10.1.1.2 with a precedence value of 5:

```
Device(config)# ipv6 access-list ipv6_Name_ACL permit ip host 10::1 host 10.1.1.2
precedence 5
```

This example shows how to create a Layer 2 MAC ACL with two permit statements. The first statement allows traffic from the host with MAC address 0001.0000.0001 to the host with MAC address 0002.0000.0001. The second statement allows only Ethertype XNS-IDP traffic from the host with MAC address 0001.0000.0002 to the host with MAC address 0002.0000.0002.

```
Device(config)# mac access-list extended maclist1
Device(config-ext-macl)# permit 0001.0000.0001 0.0.0 0002.0000.0001 0.0.0
Device(config-ext-macl)# permit 0001.0000.0002 0.0.0 0002.0000.0002 0.0.0 xns-idp
! (Note: all other access implicitly denied)
```

Examples: Classifying Traffic by Using Class Maps

This example shows how to configure the class map called *class1*. The *class1* has one match criterion, which is access list 103. It permits traffic from any host to any destination that matches a DSCP value of 10.

```
Device(config)# access-list 103 permit ip any any dscp 10
Device(config)# class-map class1
Device(config-cmap)# match access-group 103
Device(config-cmap)# end
Device#
```

This example shows how to create a class map called *class2*, which matches incoming traffic with DSCP values of 10, 11, and 12.

```
Device(config)# class-map class2
Device(config-cmap)# match ip dscp 10 11 12
Device(config-cmap)# end
Device#
```

This example shows how to create a class map called *class3*, which matches incoming traffic with IP-precedence values of 5, 6, and 7:

```
Device(config)# class-map class3
Device(config-cmap)# match ip precedence 5 6 7
Device(config-cmap)# end
Device#
```

This example shows how to configure a class map to match IP DSCP and IPv6:

```
Device(config)# Class-map cm-1
Device(config-cmap)# match ip dscp 10
Device(config-cmap)# match protocol ipv6
Device(config-cmap)# exit
Device(config)# Class-map cm-2
Device(config-cmap)# match ip dscp 20
Device(config-cmap)# match protocol ip
Device(config-cmap)# exit
Device(config)# Policy-map pml
Device(config-pmap)# class cm-1
Device(config-pmap-c)# set dscp 4
Device(config-pmap-c)# exit
Device(config-pmap)# class cm-2
Device(config-pmap-c)# set dscp 6
Device(config-pmap-c)# exit
Device(config-pmap)# exit
Device(config)# interface G1/0/1
Device(config-if)# service-policy input pml
```

This example shows how to configure a class map that applies to both IPv4 and IPv6 traffic:

```
Device(config)# ip access-list 101 permit ip any any
Device(config)# ipv6 access-list ipv6-any permit ip any any
Device(config)# Class-map cm-1
Device(config-cmap)# match access-group 101
Device(config-cmap)# exit
Device(config)# class-map cm-2
Device(config-cmap)# match access-group name ipv6-any
Device(config-cmap)# exit
Device(config)# Policy-map pml
Device(config-pmap)# class cm-1
Device(config-pmap-c)# set dscp 4
Device(config-pmap-c)# exit
Device(config-pmap)# class cm-2
Device(config-pmap-c)# set dscp 6
Device(config-pmap-c)# exit
Device(config-pmap)# exit
Device(config)# interface G0/1
Device(config-if)# switch mode access
Device(config-if)# service-policy input pml
```

Examples: Classifying, Policing, and Marking Traffic on Physical Ports Using Policy Maps

This example shows how to create a policy map and attach it to an ingress port. In the configuration, the IP standard ACL permits traffic from network 10.1.0.0. For traffic matching this classification, the DSCP value in the incoming packet is trusted. If the matched traffic exceeds an average traffic rate of 48000 b/s and a normal burst size of 8000 bytes, its DSCP is marked down (based on the policed-DSCP map) and sent:

```
Device(config)# access-list 1 permit 10.1.0.0 0.0.255.255
Device(config)# class-map ipclass1
Device(config-cmap)# match access-group 1
Device(config-cmap)# exit
Device(config)# policy-map flow1t
Device(config-pmap)# class ipclass1
Device(config-pmap-c)# trust dscp
Device(config-pmap-c)# police 1000000 8000 exceed-action policed-dscp-transmit
Device(config-pmap-c)# exit
Device(config-pmap)# exit
Device(config)# interface gigabitethernet2/0/1
Device(config-if)# service-policy input flow1t
```

This example shows how to create a Layer 2 MAC ACL with two permit statements and attach it to an ingress port. The first permit statement allows traffic from the host with MAC address 0001.0000.0001 destined for the host with MAC address 0002.0000.0001. The second permit statement allows only EtherType XNS-IDP traffic from the host with MAC address 0001.0000.0002 destined for the host with MAC address 0002.0000.0002.

```
Device(config)# mac access-list extended maclist1
Device(config-ext-mac)# permit 0001.0000.0001 0.0.0 0002.0000.0001 0.0.0
Device(config-ext-mac)# permit 0001.0000.0002 0.0.0 0002.0000.0002 0.0.0 xns-idp
Device(config-ext-mac)# exit
Device(config)# mac access-list extended maclist2
Device(config-ext-mac)# permit 0001.0000.0003 0.0.0 0002.0000.0003 0.0.0
Device(config-ext-mac)# permit 0001.0000.0004 0.0.0 0002.0000.0004 0.0.0 aarp
Device(config-ext-mac)# exit
Device(config)# class-map macclass1
Device(config-cmap)# match access-group maclist1
Device(config-cmap)# exit
Device(config)# policy-map macpolicy1
Device(config-pmap)# class macclass1
Device(config-pmap-c)# set dscp 63
Device(config-pmap-c)# exit
Device(config-pmap)# class macclass2 maclist2
Device(config-pmap-c)# set dscp 45
Device(config-pmap-c)# exit
Device(config-pmap)# exit
Device(config)# interface gigabitethernet1/0/1
Device(config-if)# mls qos trust cos
Device(config-if)# service-policy input macpolicy1
```

This example shows how to create a class map that applies to both IPv4 and IPv6 traffic with the default class applied to unclassified traffic:

```
Device(config)# ip access-list 101 permit ip any any
Device(config)# ipv6 access-list ipv6-any permit ip any any
```

```

Device(config)# class-map cm-1
Device(config-cmap)# match access-group 101
Device(config-cmap)# exit
Device(config)# class-map cm-2
Device(config-cmap)# match access-group name ipv6-any
Device(config-cmap)# exit
Device(config)# policy-map pml
Device(config-pmap)# class cm-1
Device(config-pmap-c)# set dscp 4
Device(config-pmap-c)# exit
Device(config-pmap)# class cm-2
Device(config-pmap-c)# set dscp 6
Device(config-pmap-c)# exit
Device(config-pmap)# class class-default
Device(config-pmap-c)# set dscp 10
Device(config-pmap-c)# exit
Device(config-pmap)# exit
Device(config)# interface G0/1
Device(config-if)# switch mode access
Device(config-if)# service-policy input pml

```

Examples: Classifying, Policing, and Marking Traffic on SVIs by Using Hierarchical Policy Maps

This example shows how to create a hierarchical policy map:

```

Switch> enable
Device# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Device(config)# access-list 101 permit ip any any
Device(config)# class-map cm-1
Device(config-cmap)# match access 101
Device(config-cmap)# exit
Device(config)# exit
Device#
Device#

```

This example shows how to attach the new map to an SVI:

```

Device# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Device(config)# class-map cm-interface-1
Device(config-cmap)# match input gigabitethernet3/0/1 - gigabitethernet3/0/2
Device(config-cmap)# exit
Device(config)# policy-map port-plcmap
Device(config-pmap)# class cm-interface-1
Device(config-pmap-c)# police 900000 9000 exc policed-dscp-transmit
Device(config-pmap-c)# exit
Device(config-pmap)# exit
Device(config)# policy-map vlan-plcmap
Device(config-pmap)# class cm-1
Device(config-pmap-c)# set dscp 7
Device(config-pmap-c)# service-policy port-plcmap-1
Device(config-pmap-c)# exit
Device(config-pmap)# class cm-2
Device(config-pmap-c)# service-policy port-plcmap-1
Device(config-pmap-c)# set dscp 10

```

```

Device(config-pmap) # exit
Device(config-pmap) # class cm-3
Device(config-pmap-c) # service-policy port-plcmap-2
Device(config-pmap-c) # set dscp 20
Device(config-pmap) # exit
Device(config-pmap) # class cm-4
Device(config-pmap-c) # trust dscp
Device(config-pmap) # exit
Device(config) # interface vlan 10
Device(config-if) # service-policy input vlan-plcmap
Device(config-if) # exit
Device(config) # exit
Device#

```

This example shows that when a child-level policy map is attached below a class, an action must be specified for the class:

```

Device(config) # policy-map vlan-plcmap
Device(config-pmap) # class cm-5
Device(config-pmap-c) # set dscp 7
Device(config-pmap-c) # service-policy port-plcmap-1

```

This example shows how to configure a class map to match IP DSCP and IPv6:

```

Device(config) # class-map cm-1
Device(config-cmap) # match ip dscp 10
Device(config-cmap) # match protocol ipv6
Device(config-cmap) # exit
Device(config) # class-map cm-2
Device(config-cmap) # match ip dscp 20
Device(config-cmap) # match protocol ip
Device(config-cmap) # exit
Device(config) # policy-map pml
Device(config-pmap) # class cm-1
Device(config-pmap-c) # set dscp 4
Device(config-pmap-c) # exit
Device(config-pmap) # class cm-2
Device(config-pmap-c) # set dscp 6
Device(config-pmap-c) # exit
Device(config-pmap) # exit
Device(config) # interface G1/0/1
Device(config-if) # service-policy input pml

```

This example shows how to configure default traffic class to a policy map:

```

Device# configure terminal
Device(config) # class-map cm-3
Device(config-cmap) # match ip dscp 30
Device(config-cmap) # match protocol ipv6
Device(config-cmap) # exit
Device(config) # class-map cm-4
Device(config-cmap) # match ip dscp 40
Device(config-cmap) # match protocol ip
Device(config-cmap) # exit
Device(config) # policy-map pm3
Device(config-pmap) # class class-default
Device(config-pmap) # set dscp 10
Device(config-pmap-c) # exit

```



```

Device(config-pmap)# class cm-3
Device(config-pmap-c) set dscp 4
Device(config-pmap-c) # exit
Device(config-pmap)# class cm-4
Device(config-pmap-c) # trust cos
Device(config-pmap-c) # exit
Device(config-pmap) # exit

```

This example shows how the default traffic class is automatically placed at the end of policy-map pm3 even though class-default was configured first:

```

Device# show policy-map pm3
Policy Map pm3
  Class cm-3
    set dscp 4
  Class cm-4
    trust cos
  Class class-default
    police 8000 80000 exceed-action drop
Device#

```

Examples: Classifying, Policing, and Marking Traffic by Using Aggregate Policers

This example shows how to create an aggregate policer and attach it to multiple classes within a policy map. In the configuration, the IP ACLs permit traffic from network 10.1.0.0 and from host 11.3.1.1. For traffic coming from network 10.1.0.0, the DSCP in the incoming packets is trusted. For traffic coming from host 11.3.1.1, the DSCP in the packet is changed to 56. The traffic rate from the 10.1.0.0 network and from host 11.3.1.1 is policed. If the traffic exceeds an average rate of 48000 b/s and a normal burst size of 8000 bytes, its DSCP is marked down (based on the policed-DSCP map) and sent. The policy map is attached to an ingress port.

```

Device(config)# access-list 1 permit 10.1.0.0 0.0.255.255
Device(config)# access-list 2 permit 11.3.1.1
Device(config)# mls qos aggregate-police transmit1 48000 8000 exceed-action
policed-dscp-transmit
Device(config)# class-map ipclass1
Device(config-cmap)# match access-group 1
Device(config-cmap)# exit
Device(config)# class-map ipclass2
Device(config-cmap)# match access-group 2
Device(config-cmap)# exit
Device(config)# policy-map aggflow1
Device(config-pmap)# class ipclass1
Device(config-pmap-c) # trust dscp
Device(config-pmap-c) # police aggregate transmit1
Device(config-pmap-c) # exit
Device(config-pmap)# class ipclass2
Device(config-pmap-c) # set dscp 56
Device(config-pmap-c) # police aggregate transmit1
Device(config-pmap-c) # exit
Device(config-pmap)# class class-default
Device(config-pmap-c) # set dscp 10
Device(config-pmap-c) # exit
Device(config-pmap) # exit

```

```
Device(config)# interface gigabitethernet2/0/1
Device(config-if)# service-policy input aggflow1
Device(config-if)# exit
```

Examples: Configuring DSCP Maps

This example shows how to modify and display the CoS-to-DSCP map:

```
Device(config)# mls qos map cos-dscp 10 15 20 25 30 35 40 45
Device(config)# end
Device# show mls qos maps cos-dscp
```

```
Cos-dscp map:
  cos:   0  1  2  3  4  5  6  7
-----
  dscp:  10 15 20 25 30 35 40 45
```

This example shows how to modify and display the IP-precedence-to-DSCP map:

```
Device(config)# mls qos map ip-prec-dscp 10 15 20 25 30 35 40 45
Device(config)# end
Device# show mls qos maps ip-prec-dscp
```

```
IpPrecedence-dscp map:
  ipprec:  0  1  2  3  4  5  6  7
-----
  dscp:   10 15 20 25 30 35 40 45
```

This example shows how to map DSCP 50 to 57 to a marked-down DSCP value of 0:

```
Device(config)# mls qos map policed-dscp 50 51 52 53 54 55 56 57 to 0
Device(config)# end
Device# show mls qos maps policed-dscp
```

```
Policed-dscp map:
  d1 : d2 0  1  2  3  4  5  6  7  8  9
-----
  0 :   00 01 02 03 04 05 06 07 08 09
  1 :   10 11 12 13 14 15 16 17 18 19
  2 :   20 21 22 23 24 25 26 27 28 29
  3 :   30 31 32 33 34 35 36 37 38 39
  4 :   40 41 42 43 44 45 46 47 48 49
  5 :   00 00 00 00 00 00 00 00 58 59
  6 :   60 61 62 63
```



Note In this policed-DSCP map, the marked-down DSCP values are shown in the body of the matrix. The d1 column specifies the most-significant digit of the original DSCP; the d2 row specifies the least-significant digit of the original DSCP. The intersection of the d1 and d2 values provides the marked-down value. For example, an original DSCP value of 53 corresponds to a marked-down DSCP value of 0.

This example shows how to map DSCP values 0, 8, 16, 24, 32, 40, 48, and 50 to CoS value 0 and to display the map:

```

Device(config)# mls qos map dscp-cos 0 8 16 24 32 40 48 50 to 0
Device(config)# end
Device# show mls qos maps dscp-cos
Dscp-cos map:
  d1 : d2 0  1  2  3  4  5  6  7  8  9
-----
  0 :    00 00 00 00 00 00 00 00 00 01
  1 :    01 01 01 01 01 01 00 02 02 02
  2 :    02 02 02 02 00 03 03 03 03 03
  3 :    03 03 00 04 04 04 04 04 04 04
  4 :    00 05 05 05 05 05 05 05 00 06
  5 :    00 06 06 06 06 06 07 07 07 07
  6 :    07 07 07 07

```



Note In the above DSCP-to-CoS map, the CoS values are shown in the body of the matrix. The d1 column specifies the most-significant digit of the DSCP; the d2 row specifies the least-significant digit of the DSCP. The intersection of the d1 and d2 values provides the CoS value. For example, in the DSCP-to-CoS map, a DSCP value of 08 corresponds to a CoS value of 0.

This example shows how to define the DSCP-to-DSCP-mutation map. All the entries that are not explicitly configured are not modified (remains as specified in the null map):

```

Device(config)# mls qos map dscp-mutation mutation1 1 2 3 4 5 6 7 to 0
Device(config)# mls qos map dscp-mutation mutation1 8 9 10 11 12 13 to 10
Device(config)# mls qos map dscp-mutation mutation1 20 21 22 to 20
Device(config)# mls qos map dscp-mutation mutation1 30 31 32 33 34 to 30
Device(config)# interface gigabitethernet1/0/1
Device(config-if)# mls qos trust dscp
Device(config-if)# mls qos dscp-mutation mutation1
Device(config-if)# end
Device# show mls qos maps dscp-mutation mutation1
Dscp-dscp mutation map:
mutation1:
  d1 : d2 0  1  2  3  4  5  6  7  8  9
-----
  0 :    00 00 00 00 00 00 00 00 10 10
  1 :    10 10 10 10 14 15 16 17 18 19
  2 :    20 20 20 23 24 25 26 27 28 29
  3 :    30 30 30 30 30 35 36 37 38 39
  4 :    40 41 42 43 44 45 46 47 48 49
  5 :    50 51 52 53 54 55 56 57 58 59
  6 :    60 61 62 63

```



Note In the above DSCP-to-DSCP-mutation map, the mutated values are shown in the body of the matrix. The d1 column specifies the most-significant digit of the original DSCP; the d2 row specifies the least-significant digit of the original DSCP. The intersection of the d1 and d2 values provides the mutated value. For example, a DSCP value of 12 corresponds to a mutated value of 10.

Examples: Configuring Egress Queue Characteristics

This example shows how to map DSCP values 10 and 11 to egress queue 1 and to threshold 2:

```
Device(config)# mls qos srr-queue output dscp-map queue 1 threshold 2 10 11
```

This example shows how to configure bandwidth shaping on queue 1. Because the weight ratios for queues 2, 3, and 4 are set to 0, these queues operate in shared mode. The bandwidth weight for queue 1 is 1/8, which is 12.5 percent:

```
Device(config)# interface gigabitethernet2/0/1
Device(config-if)# srr-queue bandwidth shape 8 0 0 0
```

This example shows how to configure the weight ratio of the SRR scheduler running on an egress port. Four queues are used, and the bandwidth ratio allocated for each queue in shared mode is $1/(1+2+3+4)$, $2/(1+2+3+4)$, $3/(1+2+3+4)$, and $4/(1+2+3+4)$, which is 10 percent, 20 percent, 30 percent, and 40 percent for queues 1, 2, 3, and 4. This means that queue 4 has four times the bandwidth of queue 1, twice the bandwidth of queue 2, and one-and-a-third times the bandwidth of queue 3.

```
Device(config)# interface gigabitethernet2/0/1
Device(config-if)# srr-queue bandwidth share 1 2 3 4
```

This example shows how to enable the egress expedite queue when the SRR weights are configured. The egress expedite queue overrides the configured SRR weights.

```
Device(config)# interface gigabitethernet1/0/1
Device(config-if)# srr-queue bandwidth shape 25 0 0 0
Device(config-if)# srr-queue bandwidth share 30 20 25 25
Device(config-if)# priority-queue out
Device(config-if)# end
```

This example shows how to limit the bandwidth on a port to 80 percent:

```
Device(config)# interface gigabitethernet2/0/1
Device(config-if)# srr-queue bandwidth limit 80
```

When you configure this command to 80 percent, the port is idle 20 percent of the time. The line rate drops to 80 percent of the connected speed, which is 800 Mb/s. These values are not exact because the hardware adjusts the line rate in increments of six.

Where to Go Next

Review the auto-QoS documentation to see if you can use these automated capabilities for your QoS configuration.

Additional References

Related Documents

Related Topic	Document Title
For complete syntax and usage information for the commands used in this book.	<i>Catalyst 2960-XR Switch Quality of Service Command Reference</i>

Standards and RFCs

Standard/RFC	Title
—	—

MIBs

MIB	MIBs Link
All the supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
<p>The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies.</p> <p>To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds.</p> <p>Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.</p>	http://www.cisco.com/support

Feature History and Information for QoS

Release	Modification
Cisco IOS Release 15.0(2)EX1	This feature was introduced.



CHAPTER 41

Configuring Auto-QoS

- [Prerequisites for Auto-QoS, on page 829](#)
- [Restrictions for Auto-QoS, on page 830](#)
- [Information About Configuring Auto-QoS, on page 830](#)
- [How to Configure Auto-QoS, on page 833](#)
- [Monitoring Auto-QoS, on page 837](#)
- [Configuration Examples for Auto-QoS, on page 838](#)
- [Where to Go Next for Auto-QoS, on page 846](#)
- [Additional References, on page 846](#)
- [Feature History and Information for Auto-QoS, on page 847](#)

Prerequisites for Auto-QoS

Before configuring standard QoS or auto-QoS, you must have a thorough understanding of these items:

- The types of applications used and the traffic patterns on your network.
- Traffic characteristics and needs of your network. Is the traffic bursty? Do you need to reserve bandwidth for voice and video streams?
- Bandwidth requirements and speed of the network.
- Location of congestion points in the network.

Auto-QoS VoIP Considerations

Before configuring auto-QoS for VoIP, you should be aware of this information:

- Auto-QoS configures the switch for VoIP with Cisco IP Phones on nonrouted and routed ports. Auto-QoS also configures the switch for VoIP with devices running the Cisco SoftPhone application.



Note When a device running Cisco SoftPhone is connected to a nonrouted or routed port, the switch supports only one Cisco SoftPhone application per port.

- When enabling auto-QoS with a Cisco IP Phone on a routed port, you must assign a static IP address to the IP phone.
- This release supports only Cisco IP SoftPhone Version 1.3(3) or later.
- Connected devices must use Cisco Call Manager Version 4 or later.
- Auto-QoS VoIP uses the **priority-queue** interface configuration command for an egress interface. You can also configure a policy-map and trust device on the same interface for Cisco IP phones.

Auto-QoS Enhanced Considerations

Auto-QoS is enhanced to support video. Automatic configurations are generated that classify and trust traffic from Cisco TelePresence systems and Cisco IP cameras.

Before configuring auto-QoS enhanced, you should be aware of this information:

- The **auto qos srnd4** global configuration command is generated as a result of enhanced auto-QoS configuration.

Restrictions for Auto-QoS

The following are restrictions for automatic QoS (auto-QoS):

- After auto-QoS is enabled, do not modify a policy map or aggregate policer that includes *AutoQoS* in its name. If you need to modify the policy map or aggregate policer, make a copy of it, and change the copied policy map or policer. To use this new policy map instead of the generated one, remove the generated policy map from the interface, and apply the new policy map to the interface.
- To take advantage of the auto-QoS defaults, you should enable auto-QoS before you configure other QoS commands. If necessary, you can fine-tune the QoS configuration, but we recommend that you do so only after the auto-QoS configuration is completed.
- By default, the CDP is enabled on all ports. For auto-QoS to function properly, do not disable CDP.



Note You can enable auto-QoS on static, dynamic-access, voice VLAN access, and trunk ports.

Information About Configuring Auto-QoS

Auto-QoS Overview

You can use the auto-QoS feature to simplify the deployment of QoS features. Auto-QoS determines the network design and enables QoS configurations so that the switch can prioritize different traffic flows. It uses the egress queues instead of using the default (disabled) QoS behavior. The switch offers best-effort service to each packet, regardless of the packet contents or size, and sends it from a single queue.

When you enable auto-QoS, it automatically classifies traffic based on the traffic type and ingress packet label. The switch uses the classification results to choose the appropriate egress queue.

You can use auto-QoS commands to identify ports connected to the following Cisco devices:

- Cisco IP Phones
- Devices running the Cisco SoftPhone application
- Cisco TelePresence
- Cisco IP Camera
- Cisco digital media player

You also use the auto-QoS commands to identify ports that receive trusted traffic through an uplink. Auto-QoS then performs these functions:

- Detects the presence or absence of auto-QoS devices through conditional trusted interfaces.
- Configures QoS classification
- Configures egress queues

Auto-QoS Compact Overview

When you enter an auto-QoS command, the switch displays all the generated commands as if the commands were entered from the CLI. You can use the auto-QoS compact feature to hide the auto-QoS generated commands from the running configuration. This would make it easier to comprehend the running-configuration and also help to increase efficient usage of memory.

Generated Auto-QoS Configuration

By default, auto-QoS is disabled on all ports. Packets are not modified--the CoS, DSCP and IP precedence values in the packet are not changed.

When you enable the auto-QoS feature on the first port of the interface:

- Ingress packet label is used to categorize traffic, to assign packet labels, and to configure the ingress and egress queues.
- QoS is globally enabled (**mls qos** global configuration command), and other global configuration commands are automatically generated. (See [Examples: Global Auto-QoS Configuration, on page 838](#)).
- Switch enables the trusted boundary feature and uses the Cisco Discovery Protocol (CDP) to detect the presence of a supported device.
- Policing is used to determine whether a packet is in or out of profile and specifies the action on the packet.

VoIP Device Specifics

The following actions occur when you issue these auto-QoS commands on a port:

- **auto qos voip cisco-phone**—When you enter this command on a port at the network edge connected to a Cisco IP Phone, the switch enables the trusted boundary feature. If the packet does not have a DSCP value of 24, 26, or 46 or is out of profile, the switch changes the DSCP value to 0. When there is no

Cisco IP Phone, the ingress classification is set to not trust the QoS label in the packet. The policing is applied to the traffic matching the policy-map classification before the switch enables the trust boundary feature.

- **auto qos voip cisco-softphone**—When you enter this interface configuration command on a port at the network edge that is connected to a device running the Cisco SoftPhone, the switch uses policing to determine whether a packet is in or out of profile and to specify the action on the packet. If the packet does not have a DSCP value of 24, 26, or 46 or is out of profile, the switch changes the DSCP value to 0.
- **auto qos voip trust**—When you enter this interface configuration command on a port connected to the network interior, the switch trusts the CoS value for nonrouted ports or the DSCP value for routed ports in ingress packets (the assumption is that traffic has already been classified by other edge devices).

The switch configures egress queues on the port according to the settings in the following tables.

Table 87: Traffic Types, Packet Labels, and Queues

	VoIP Data Traffic	VoIP Control Traffic	Routing Protocol Traffic	STP BPDU Traffic	Real-Time Video Traffic	All Other Traffic	
DSCP value	46	24, 26	48	56	34	–	
CoS value	5	3	6	7	3	–	
CoS-to-Egress queue map	4, 5 (queue 1)	2, 3, 6, 7 (queue 2)			0 (queue 3)	2 (queue 3)	0, 1 (queue 4)

The following table shows the generated auto-QoS configuration for the egress queues.

Table 88: Auto-QoS Configuration for the Egress Queues

Egress Queue	Queue Number	CoS-to-Queue Map	Queue Weight (Bandwidth)	Queue (Buffer) Size for Gigabit-Capable Ports	Queue (Buffer) Size for 10/100 Ethernet Ports
Priority	1	4, 5	Up to 100 percent	25 percent	15 percent
SRR shared	2	2, 3, 6, 7	10 percent	25 percent	25 percent
SRR shared	3	0	60 percent	25 percent	40 percent
SRR shared	4	1	20 percent	25 percent	20 percent

- When you enable auto-QoS by using the **auto qos voip cisco-phone**, the **auto qos voip cisco-softphone**, or the **auto qos voip trust** interface configuration command, the switch automatically generates a QoS configuration based on the traffic type and ingress packet label and applies the commands listed in [Examples: Global Auto-QoS Configuration, on page 838](#) to the port.

Effects of Auto-QoS on Running Configuration

When auto-QoS is enabled, the **auto qos** interface configuration commands and the generated global configuration are added to the running configuration.

The switch applies the auto-QoS-generated commands as if the commands were entered from the CLI. An existing user configuration can cause the application of the generated commands to fail or to be overridden by the generated commands. These actions may occur without warning. If all the generated commands are successfully applied, any user-entered configuration that was not overridden remains in the running configuration. Any user-entered configuration that was overridden can be retrieved by reloading the switch without saving the current configuration to memory. If the generated commands are not applied, the previous running configuration is restored.

Effects of Auto-QoS Compact on Running Configuration

If auto-QoS compact is enabled:

- Only the auto-QoS commands entered from the CLI are displayed in running-config.
- The generated global and interface configurations are hidden.
- When you save the configuration, only the auto-qos commands you have entered are saved (and not the hidden configuration).
- When you reload the switch, the system detects and re-executes the saved auto-QoS commands and the AutoQoS SRND4.0 compliant config-set is generated .



Note Do not make changes to the auto-QoS-generated commands when auto-QoS compact is enabled, because user-modifications are overridden when the switch reloads.

When auto-qos global compact is enabled:

- **show derived-config** command can be used to view hidden AQC derived commands.
- AQC commands will not be stored to memory. They will be regenerated every time the switch is reloaded.
- When compaction is enabled, auto-qos generated commands should not be modified .
- If the interface is configured with auto-QoS and if AQC needs to be disabled, auto-qos should be disabled at interface level first.

How to Configure Auto-QoS

Configuring Auto-QoS

Enabling Auto-QoS

For optimum QoS performance, enable auto-QoS on all the devices in your network.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. Use one of the following:
 - **auto qos voip** {**cisco-phone** | **cisco-softphone** | **trust**}
 - **auto qos video** {**cts** | **ip-camera** | **media-player**}
 - **auto qos classify** [**police**]
 - **auto qos trust** {**cos** | **dscp**}
4. **exit**
5. **interface** *interface-id*
6. **auto qos trust**
7. **end**
8. **show auto qos interface** *interface-id*

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 2	interface <i>interface-id</i> Example: Device(config)# interface gigabitethernet 3/0/1	Specifies the port that is connected to a video device or the uplink port that is connected to another trusted switch or router in the network interior, and enters interface configuration mode.
Step 3	Use one of the following: <ul style="list-style-type: none"> • auto qos voip {cisco-phone cisco-softphone trust} • auto qos video {cts ip-camera media-player} • auto qos classify [police] • auto qos trust {cos dscp} Example: Device(config-if)# auto qos trust dscp	Enables auto-QoS for VoIP. <ul style="list-style-type: none"> • cisco-phone—If the port is connected to a Cisco IP Phone, the QoS labels of incoming packets are trusted only when the telephone is detected. • cisco-softphone—The port is connected to device running the Cisco SoftPhone feature. • trust—The uplink port is connected to a trusted switch or router, and the VoIP traffic classification in the ingress packet is trusted. Enables auto-QoS for a video device. <ul style="list-style-type: none"> • cts—A port connected to a Cisco Telepresence system. • ip-camera—A port connected to a Cisco video surveillance camera.

	Command or Action	Purpose
		<ul style="list-style-type: none"> • media-player—A port connected to a CDP-capable Cisco digital media player. <p>QoS labels of incoming packets are trusted only when the system is detected.</p> <p>Enables auto-QoS for classification.</p> <ul style="list-style-type: none"> • police—Policing is set up by defining the QoS policy maps and applying them to ports (port-based QoS). <p>Enables auto-QoS for trusted interfaces.</p> <ul style="list-style-type: none"> • cos—Class of service. • dscp—Differentiated Services Code Point. • <cr>—Trust interface. <p>Note To view a list of commands that are automatically generated by issuing one of the auto-QoS commands listed here, you need to be in debug mode. Refer to the <i>Catalyst 2960-XR Switch QoS Command Reference Guide</i> for examples of how to run the appropriate debug command to view a list of these commands.</p>
Step 4	<p>exit</p> <p>Example:</p> <pre>Device(config-if) # exit</pre>	Returns to global configuration mode.
Step 5	<p>interface interface-id</p> <p>Example:</p> <pre>Device(config) # interface gigabitethernet 2/0/1</pre>	Specifies the switch port identified as connected to a trusted switch or router, and enters interface configuration mode.
Step 6	<p>auto qos trust</p> <p>Example:</p> <pre>Device(config-if) # auto qos trust</pre>	Enables auto-QoS on the port, and specifies that the port is connected to a trusted router or switch.
Step 7	<p>end</p> <p>Example:</p> <pre>Device(config-if) # end</pre>	Returns to privileged EXEC mode.

	Command or Action	Purpose
Step 8	show auto qos interface <i>interface-id</i> Example: Device# show auto qos interface gigabitethernet 2/0/1	Verifies your entries. This command displays the auto-QoS command on the interface on which auto-QoS was enabled. You can use the show running-config privileged EXEC command to display the auto-QoS configuration and the user modifications.

Enabling Auto-Qos Compact

To enable auto-Qos compact, enter this command:

SUMMARY STEPS

1. **configure terminal**
2. **auto qos global compact**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 2	auto qos global compact Example: Device(config)# auto qos global compact	Enables auto-Qos compact and generates (hidden) the global configurations for auto-QoS. You can then enter the auto-QoS command you want to configure in the interface configuration mode and the interface commands that the system generates are also hidden. To display the auto-QoS configuration that has been applied, use these the privileged EXEC commands: <ul style="list-style-type: none"> • show derived-config • show policy-map • show access-list • show class-map • show table-map • show auto-qos • show policy-map interface • show ip access-lists These commands will have keyword " AutoQos- ".

What to do next

To disable auto-QoS compact, remove auto-QoS instances from all interfaces by entering the **no** form of the corresponding auto-QoS commands and then enter the **no auto qos global compact** global configuration command.

Troubleshooting Auto-QoS

To display the QoS commands that are automatically generated when auto-QoS is enabled or disabled, enter the **debug auto qos** privileged EXEC command before you enable auto-QoS. For more information, see the **debug auto qos** command in the command reference for this release.

To disable auto-QoS on a port, use the **no** form of the **auto qos** command interface configuration command, such as **no auto qos voip**.



Note Auto-QoS generated global commands can also be removed manually if desired.

Only the auto-QoS-generated interface configuration commands for this port are removed. If this is the last port on which auto-QoS is enabled and you enter the **no auto qos voip** command, auto-QoS is considered disabled even though the auto-QoS-generated global configuration commands remain (to avoid disrupting traffic on other ports affected by the global configuration).

You can use the **no mls qos** global configuration command to disable the auto-QoS-generated global configuration commands. With QoS disabled, there is no concept of trusted or untrusted ports because the packets are not modified (the CoS, DSCP, and IP precedence values in the packet are not changed). Traffic is switched in pass-through mode (packets are switched without any rewrites and classified as best effort without any policing).

Monitoring Auto-QoS

Table 89: Commands for Monitoring Auto-QoS

Command	Description
show auto qos [interface <i>interface-type</i>]	Displays the initial auto-QoS configuration. You can compare the show auto qos and the show running-config command output to identify the user-defined QoS settings.
show mls qos [aggregate policer interface maps queue-set stack-port stack-qset vlan]	Displays information about the QoS configuration that might be affected by auto-QoS.
show mls qos aggregate policer <i>policer_name</i>	Displays information about the QoS aggregate policer configuration that might be affected by auto-QoS.
show mls qos interface [<i>interface-type</i> buffers policers queueing statistics]	Displays information about the QoS interface configuration that might be affected by auto-QoS.

Command	Description
show mls qos maps [cos-dscp cos-output-q dscp-cos dscp-mutation dscp-output-q ip-prec-dscp policed-dscp]	Displays information about the QoS maps configuration that might be affected by auto-QoS.
show mls qos queue-set <i>queue-set ID</i>	Displays information about the QoS queue-set configuration that might be affected by auto-QoS.
show mls qos stack-port buffers	Displays information about the QoS stack port buffer configuration that might be affected by auto-QoS.
show mls qos stack-qset	Displays information about the QoS stack queue set configuration that might be affected by auto-QoS.
show running-config	Displays information about the QoS configuration that might be affected by auto-QoS. You can compare the show auto qos and the show running-config command output to identify the user-defined QoS settings.

Configuration Examples for Auto-QoS

Examples: Global Auto-QoS Configuration

The following table describes the automatically generated commands for auto-QoS and enhanced auto-QoS by the switch.

Table 90: Generated Auto-QoS Configuration

Description	Automatically Generated Command {voip}	Enhanced Automatically Generated Command {Video Trust Classify}
The switch automatically enables standard QoS and configures the CoS-to-DSCP map (maps CoS values in incoming packets to a DSCP value).	Device(config)# mls qos Device(config)# mls qos map cos-dscp 0 8 16 26 32 46 48 56	Device(config)# mls qos Device(config)# mls qos map cos-dscp 0 8 16 24 32 46 48 56

Description	Automatically Generated Command {voip}	Enhanced Automatically Generated Command {Video Trust Classify}
<p>The switch automatically maps CoS values to an egress queue and to a threshold ID.</p>	<pre> Device(config)# no mls qos srr-queue output cos-map Device(config)# mls qos srr-queue output cos-map queue 1 threshold 3 5 Device(config)# mls qos srr-queue output cos-map queue 2 threshold 3 3 6 7 Device(config)# mls qos srr-queue output cos-map queue 3 threshold 3 2 4 Device(config)# mls qos srr-queue output cos-map queue 4 threshold 2 1 Device(config)# mls qos srr-queue output cos-map queue 4 threshold 3 0 </pre>	<pre> Device(config)# no mls qos srr-queue output cos-map Device(config)# mls qos srr-queue output cos-map queue 1 threshold 3 4 5 Device(config)# mls qos srr-queue output cos-map queue 2 threshold 3 6 7 Device(config)# mls qos srr-queue output cos-map queue 2 threshold 1 2 Device(config)# mls qos srr-queue output cos-map queue 2 threshold 2 3 Device(config)# mls qos srr-queue output cos-map queue 3 threshold 3 0 Device(config)# mls qos srr-queue output cos-map queue 4 threshold 3 1 </pre>

Description	Automatically Generated Command {voip}	Enhanced Automatically Generated Command {Video Trust Classify}
<p>The switch automatically maps DSCP values to an egress queue and to a threshold ID.</p>	<pre> Device(config)# no mls qos srr-queue output dscp-map Device(config)# mls qos srr-queue output dscp-map queue 1 threshold 3 40 41 42 43 44 45 46 47 Device(config)# mls qos srr-queue output dscp-map queue 2 threshold 3 24 25 26 27 28 29 30 31 Device(config)# mls qos srr-queue output dscp-map queue 2 threshold 3 48 49 50 51 52 53 54 55 Device(config)# mls qos srr-queue output dscp-map queue 2 threshold 3 56 57 58 59 60 61 62 63 Device(config)# mls qos srr-queue output dscp-map queue 3 threshold 3 16 17 18 19 20 21 22 23 Device(config)# mls qos srr-queue output dscp-map queue 3 threshold 3 32 33 34 35 36 37 38 39 Device(config)# mls qos srr-queue output dscp-map queue 4 threshold 1 8 Device(config)# mls qos srr-queue output dscp-map queue 4 threshold 2 9 10 11 12 13 14 15 Device(config)# mls qos srr-queue output dscp-map queue 4 threshold 3 0 1 2 3 4 5 6 7 </pre>	<pre> Device(config)# no mls qos srr-queue output dscp-map Device(config)# mls qos srr-queue output dscp-map queue 1 threshold 3 32 33 40 41 42 43 44 45 46 47 Device(config)# mls qos srr-queue output dscp-map queue 2 threshold 1 16 17 18 19 20 21 22 23 Device(config)# mls qos srr-queue output dscp-map queue 2 threshold 1 26 27 28 29 30 31 34 35 36 37 38 39 Device(config)# mls qos srr-queue output dscp-map queue 2 threshold 2 24 Device(config)# mls qos srr-queue output dscp-map queue 2 threshold 3 48 49 50 51 52 53 54 55 56 Device(config)# mls qos srr-queue output dscp-map queue 2 threshold 3 57 58 59 60 61 62 63 Device(config)# mls qos srr-queue output dscp-map queue 3 threshold 3 0 1 2 3 4 5 6 7 Device(config)# mls qos srr-queue output dscp-map queue 4 threshold 1 8 9 11 13 15 Device(config)# mls qos srr-queue output dscp-map queue 4 threshold 2 10 12 14 </pre>

Description	Automatically Generated Command {voip}	Enhanced Automatically Generated Command {Video Trust Classify}
<p>The switch automatically configures the egress queue buffer sizes. It configures the bandwidth and the SRR mode (shaped or shared) on the egress queues mapped to the port.</p>	<pre>Device(config)# mls qos queue-set output 1 threshold 1 138 138 92 138 Device(config)# mls qos queue-set output 1 threshold 2 138 138 92 400 Device(config)# mls qos queue-set output 1 threshold 3 36 77 100 318 Device(config)# mls qos queue-set output 1 threshold 4 20 50 67 400 Device(config)# mls qos queue-set output 2 threshold 1 149 149 100 149 Device(config)# mls qos queue-set output 2 threshold 2 118 118 100 235 Device(config)# mls qos queue-set output 2 threshold 3 41 68 100 272 Device(config)# mls qos queue-set output 2 threshold 4 42 72 100 242 Device(config)# mls qos queue-set output 1 buffers 10 10 26 54 Device(config)# mls qos queue-set output 2 buffers 16 6 17 61 Device(config-if)# priority-queue out Device(config-if)# srr-queue bandwidth share 10 10 60 20</pre>	<pre>Device(config)# mls qos queue-set output 1 threshold 2 100 100 50 200 Device(config)# mls qos queue-set output 1 threshold 2 125 125 100 400 Device(config)# mls qos queue-set output 1 threshold 3 100 100 100 400 Device(config)# mls qos queue-set output 1 threshold 4 60 150 50 200 Device(config)# mls qos queue-set output 1 buffers 15 25 40 20</pre>

Examples: Auto-QoS Generated Configuration for VoIP Devices

If you entered the **auto qos voip cisco-phone** command, the switch automatically enables the trusted boundary feature, which uses the CDP to detect the presence or absence of a Cisco IP Phone.

```
Device(config-if)# mls qos trust device cisco-phone
```

If you entered the **auto qos voip cisco-softphone** command, the switch automatically creates class maps and policy maps.

```
Device(config)# mls qos map policed-dscp 24 26 46 to 0
Device(config)# class-map match-all AutoQoS-VoIP-RTP-Trust
Device(config-cmap)# match ip dscp ef
Device(config)# class-map match-all AutoQoS-VoIP-Control-Trust
Device(config-cmap)# match ip dscp cs3 af31
Device(config)# policy-map AutoQoS-Police-SoftPhone
Device(config-pmap)# class AutoQoS-VoIP-RTP-Trust
Device(config-pmap-c)# set dscp ef
Device(config-pmap-c)# police 320000 8000 exceed-action policed-dscp-transmit
Device(config-pmap)# class AutoQoS-VoIP-Control-Trust
Device(config-pmap-c)# set dscp cs3
Device(config-pmap-c)# police 32000 8000 exceed-action policed-dscp-transmit
```

After creating the class maps and policy maps, the switch automatically applies the policy map called *AutoQoS-Police-SoftPhone* to an ingress interface on which auto-QoS with the Cisco SoftPhone feature is enabled.

```
Device(config-if)#service-policy input AutoQoS-Police-SoftPhone
```

If you entered the **auto qos voip cisco-phone** command, the switch automatically creates class maps and policy maps.

```
Device(config-if)# mls qos trust device cisco-phone
```

If you entered the **auto qos voip cisco-softphone** command, the switch automatically creates class maps and policy maps.

```
Device(config)# mls qos map policed-dscp 24 26 46 to 0
Device(config)# class-map match-all AutoQoS-VoIP-RTP-Trust
Device(config-cmap)# match ip dscp ef
Device(config)# class-map match-all AutoQoS-VoIP-Control-Trust
Device(config-cmap)# match ip dscp cs3 af31
Device(config)# policy-map AutoQoS-Police-CiscoPhone
Device(config-pmap)# class AutoQoS-VoIP-RTP-Trust
Device(config-pmap-c)# set dscp ef
Device(config-pmap-c)# police 320000 8000 exceed-action policed-dscp-transmit
Device(config-pmap)# class AutoQoS-VoIP-Control-Trust
Device(config-pmap-c)# set dscp cs3
Device(config-pmap-c)# police 32000 8000 exceed-action policed-dscp-transmit
```

After creating the class maps and policy maps, the switch automatically applies the policy map called *AutoQoS-Police-SoftPhone* to an ingress interface on which auto-QoS with the Cisco SoftPhone feature is enabled.

```
Device(config-if)#service-policy input AutoQoS-Police-SoftPhone
```

Examples: Auto-QoS Generated Configuration For Enhanced Video, Trust, and Classify Devices

If you entered the following enhanced auto-QoS commands, the switch configures a CoS-to-DSCP map (maps CoS values in incoming packets to a DSCP value):

- **auto qos video cts**
- **auto qos video ip-camera**
- **auto qos video media-player**
- **auto qos trust**
- **auto qos trust cos**
- **auto qos trust dscp**

The following command is initiated after entering one of the above auto-QoS commands:

```
Device(config)# mls qos map cos-dscp 0 8 16 24 32 46 48 56
```



Note No class maps and policy maps are configured.

If you entered the **auto qos classify** command, the switch automatically creates class maps and policy maps (as shown below).

```
Device(config)# mls qos map policed-dscp 0 10 18 24 26 46 to 8
Device(config)# mls qos map cos-dscp 0 8 16 24 32 46 48 56
Device(config)# class-map match-all AUTOQOS_MULTIENTHANCED_CONF_CLASS
Device(config-cmap)# match access-group name AUTOQOS-ACL-MULTIENTHANCED-CONF
Device(config)# class-map match-all AUTOQOS_DEFAULT_CLASS
Device(config-cmap)# match access-group name AUTOQOS-ACL-DEFAULT
Device(config)# class-map match-all AUTOQOS_TRANSACTION_CLASS
Device(config-cmap)# match access-group name AUTOQOS-ACL-TRANSACTIONAL-DATA
Device(config)# class-map match-all AUTOQOS_SIGNALING_CLASS
Device(config-cmap)# match access-group name AUTOQOS-ACL-SIGNALING
Device(config)# class-map match-all AUTOQOS_BULK_DATA_CLASS
Device(config-cmap)# match access-group name AUTOQOS-ACL-BULK-DATA
Device(config)# class-map match-all AUTOQOS_SCAVANGER_CLASS
Device(config-cmap)# match access-group name AUTOQOS-ACL-SCAVANGER
Device(config)# policy-map AUTOQOS-SRND4-CLASSIFY-POLICY
Device(config-pmap)# class AUTOQOS_MULTIENTHANCED_CONF_CLASS
Device(config-pmap-c)# set dscp af41
Device(config-pmap)# class AUTOQOS_BULK_DATA_CLASS
Device(config-pmap-c)# set dscp af11
Device(config-pmap)# class AUTOQOS_TRANSACTION_CLASS
Device(config-pmap-c)# set dscp af21
Device(config-pmap)# class AUTOQOS_SCAVANGER_CLASS
Device(config-pmap-c)# set dscp cs1
Device(config-pmap)# class AUTOQOS_SIGNALING_CLASS
Device(config-pmap-c)# set dscp cs3
Device(config-pmap)# class AUTOQOS_DEFAULT_CLASS
Device(config-pmap-c)# set dscp default
;
```

```
Device(config-if)# service-policy input AUTOQOS-SRND4-CLASSIFY-POLICY
```

If you entered the **auto qos classify police** command, the switch automatically creates class maps and policy maps (as shown below).

```
Device(config)# mls qos map policed-dscp 0 10 18 24 26 46 to 8
Device(config)# mls qos map cos-dscp 0 8 16 24 32 46 48 56
Device(config)# class-map match-all AUTOQOS_MULTITIENHANCED_CONF_CLASS
Device(config-cmap)# match access-group name AUTOQOS-ACL-MULTITIENHANCED-CONF
Device(config)# class-map match-all AUTOQOS_DEFAULT_CLASS
Device(config-cmap)# match access-group name AUTOQOS-ACL-DEFAULT
Device(config-cmap)# class-map match-all AUTOQOS_TRANSACTION_CLASS
Device(config-cmap)# match access-group name AUTOQOS-ACL-TRANSACTIONAL-DATA
Device(config)# class-map match-all AUTOQOS_SIGNALING_CLASS
Device(config-cmap)# match access-group name AUTOQOS-ACL-SIGNALING
Device(config)# class-map match-all AUTOQOS_BULK_DATA_CLASS
Device(config-cmap)# match access-group name AUTOQOS-ACL-BULK-DATA
Device(config)# class-map match-all AUTOQOS_SCAVANGER_CLASS
Device(config-cmap)# match access-group name AUTOQOS-ACL-SCAVANGER
Device(config)# policy-map AUTOQOS-SRND4-CLASSIFY-POLICE-POLICY
Device(config-pmap)# class AUTOQOS_MULTITIENHANCED_CONF_CLASS
Device(config-pmap-c)# set dscp af41
Device(config-pmap-c)# police 5000000 8000 exceed-action drop
Device(config-pmap)# class AUTOQOS_BULK_DATA_CLASS
Device(config-pmap-c)# set dscp af11
Device(config-pmap-c)# police 10000000 8000 exceed-action policed-dscp-transmit
Device(config-pmap)# class AUTOQOS_TRANSACTION_CLASS
Device(config-pmap-c)# set dscp af21
Device(config-pmap-c)# police 10000000 8000 exceed-action policed-dscp-transmit
Device(config-pmap)# class AUTOQOS_SCAVANGER_CLASS
Device(config-pmap-c)# set dscp cs1
Device(config-pmap-c)# police 10000000 8000 exceed-action drop
Device(config-pmap)# class AUTOQOS_SIGNALING_CLASS
Device(config-pmap-c)# set dscp cs3
Device(config-pmap-c)# police 32000 8000 exceed-action drop
Device(config-pmap)# class AUTOQOS_DEFAULT_CLASS
Device(config-pmap-c)# set dscp default
Device(config-pmap-c)# police 10000000 8000 exceed-action policed-dscp-transmit
;
Device(config-if)# service-policy input AUTOQOS-SRND4-CLASSIFY-POLICE-POLICY
```

This is the enhanced configuration for the **auto qos voip cisco-phone** command:

```
Device(config)# mls qos map policed-dscp 0 10 18 24 26 46 to 8
Device(config)# mls qos map cos-dscp 0 8 16 24 32 46 48 56
Device(config)# class-map match-all AUTOQOS_VOIP_DATA_CLASS
Device(config-cmap)# match ip dscp ef
Device(config)# class-map match-all AUTOQOS_DEFAULT_CLASS
Device(config-cmap)# match access-group name AUTOQOS-ACL-DEFAULT
Device(config)# class-map match-all AUTOQOS_VOIP_SIGNAL_CLASS
Device(config-cmap)# match ip dscp cs3
Device(config)# policy-map AUTOQOS-SRND4-CISCOPHONE-POLICY
Device(config-pmap)# class AUTOQOS_VOIP_DATA_CLASS
Device(config-pmap-c)# set dscp ef
Device(config-pmap-c)# police 128000 8000 exceed-action policed-dscp-transmit
Device(config-pmap)# class AUTOQOS_VOIP_SIGNAL_CLASS
Device(config-pmap-c)# set dscp cs3
Device(config-pmap-c)# police 32000 8000 exceed-action policed-dscp-transmit
Device(config-pmap)# class AUTOQOS_DEFAULT_CLASS
```

```
Device(config-pmap-c) # set dscp default
Device(config-pmap-c) # police 10000000 8000 exceed-action policed-dscp-transmit
;
Device(config-if) # service-policy input AUTOQOS-SRND4-CISCOPHONE-POLICY
```

This is the enhanced configuration for the **auto qos voip cisco-softphone** command:

```
Device(config)# mls qos map policed-dscp 0 10 18 24 26 46 to 8
Device(config)# mls qos map cos-dscp 0 8 16 24 32 46 48 56
Device(config)# class-map match-all AUTOQOS_MULTIHANCED_CONF_CLASS
Device(config-cmap)# match access-group name AUTOQOS-ACL-MULTIHANCED-CONF
Device(config)# class-map match-all AUTOQOS_VOIP_DATA_CLASS
Device(config-cmap)# match ip dscp ef
Device(config)# class-map match-all AUTOQOS_DEFAULT_CLASS
Device(config-cmap)# match access-group name AUTOQOS-ACL-DEFAULT
Device(config)# class-map match-all AUTOQOS_TRANSACTION_CLASS
Device(config-cmap)# match access-group name AUTOQOS-ACL-TRANSACTIONAL-DATA
Device(config)# class-map match-all AUTOQOS_VOIP_SIGNAL_CLASS
Device(config-cmap)# match ip dscp cs3
Device(config)# class-map match-all AUTOQOS_SIGNALING_CLASS
Device(config-cmap)# match access-group name AUTOQOS-ACL-SIGNALING
Device(config)# class-map match-all AUTOQOS_BULK_DATA_CLASS
Device(config-cmap)# match access-group name AUTOQOS-ACL-BULK-DATA
Device(config)# class-map match-all AUTOQOS_SCAVANGER_CLASS
Device(config-cmap)# match access-group name AUTOQOS-ACL-SCAVANGER

Device(config)# policy-map AUTOQOS-SRND4-SOFTPHONE-POLICY
Device(config-pmap)# class AUTOQOS_VOIP_DATA_CLASS
Device(config-pmap-c)# set dscp ef
Device(config-pmap-c)# police 128000 8000 exceed-action policed-dscp-transmit
Device(config-pmap)# class AUTOQOS_VOIP_SIGNAL_CLASS
Device(config-pmap-c)# set dscp cs3
Device(config-pmap-c)# police 32000 8000 exceed-action policed-dscp-transmit
Device(config-pmap)# class AUTOQOS_MULTIHANCED_CONF_CLASS
Device(config-pmap-c)# set dscp af41
Device(config-pmap-c)# police 5000000 8000 exceed-action drop
Device(config-pmap)# class AUTOQOS_BULK_DATA_CLASS
Device(config-pmap-c)# set dscp af11
Device(config-pmap-c)# police 10000000 8000 exceed-action policed-dscp-transmit
Device(config-pmap)# class AUTOQOS_TRANSACTION_CLASS
Device(config-pmap-c)# set dscp af21
Device(config-pmap-c)# police 10000000 8000 exceed-action policed-dscp-transmit
Device(config-pmap)# class AUTOQOS_SCAVANGER_CLASS
Device(config-pmap-c)# set dscp cs1
Device(config-pmap-c)# police 10000000 8000 exceed-action drop
Device(config-pmap)# class AUTOQOS_SIGNALING_CLASS
Device(config-pmap-c)# set dscp cs3
Device(config-pmap-c)# police 32000 8000 exceed-action drop
Device(config-pmap)# class AUTOQOS_DEFAULT_CLASS
Device(config-pmap-c)# set dscp default
;
Device(config-if) # service-policy input AUTOQOS-SRND4-SOFTPHONE-POLICY
```

auto qos global compact

The following is an example of the **auto qos global compact** command.

```
Device# configure terminal
```

```

Device(config)# auto qos global compact
Device(config)# interface GigabitEthernet1/2
Device(config-if)# auto qos voip cisco-phone

Device# show auto-qos

GigabitEthernet1/2
auto qos voip cisco-phone

Device# show running-config interface GigabitEthernet 1/0/2

interface GigabitEthernet1/0/2
auto qos voip cisco-phone
end

```

Where to Go Next for Auto-QoS

Review the QoS documentation if you require any specific QoS changes to your auto-QoS configuration.

Additional References

Related Documents

Related Topic	Document Title
For complete syntax and usage information for the commands used in this book.	<i>Catalyst 2960-XR Switch Quality of Service Command Reference</i>

Standards and RFCs

Standard/RFC	Title
—	—

MIBs

MIB	MIBs Link
All the supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
<p>The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies.</p> <p>To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds.</p> <p>Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.</p>	http://www.cisco.com/support

Feature History and Information for Auto-QoS

Release	Modification
Cisco IOS Release 15.0(2)EX1	This feature was introduced.



PART **X**

Routing

- [Configuring IP Unicast Routing, on page 851](#)
- [Configuring Bidirectional Forwarding Detection, on page 923](#)
- [Bidirectional Forwarding Detection, on page 935](#)
- [Configuring IPv6 First Hop Security, on page 947](#)
- [Routing Information Protocol, on page 981](#)



CHAPTER 42

Configuring IP Unicast Routing

- [Information About Configuring IP Unicast Routing, on page 851](#)
- [Information About IP Routing, on page 852](#)
- [How to Configure IP Routing, on page 858](#)
- [How to Configure IP Addressing, on page 859](#)
- [Monitoring and Maintaining IP Addressing, on page 876](#)
- [How to Configure IP Unicast Routing, on page 877](#)
- [Information About RIP, on page 878](#)
- [How to Configure RIP, on page 879](#)
- [Configuration Example for Summary Addresses and Split Horizon, on page 886](#)
- [Information About OSPF, on page 886](#)
- [How to Configure OSPF, on page 889](#)
- [Monitoring OSPF, on page 899](#)
- [Configuration Examples for OSPF, on page 900](#)
- [Information About EIGRP, on page 900](#)
- [Configuring Unicast Reverse Path Forwarding, on page 901](#)
- [Protocol-Independent Features, on page 901](#)
- [Monitoring and Maintaining the IP Network, on page 921](#)

Information About Configuring IP Unicast Routing

This module describes how to configure IP Version 4 (IPv4) unicast routing on the switch.

A switch stack operates and appears as a single router to the rest of the routers in the network. Basic routing functions like static routing are available with IP Lite both the IP Base feature set and the IP Services feature set.



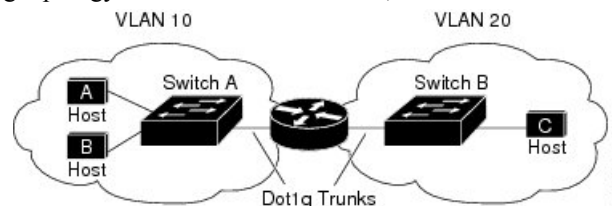
Note In addition to IPv4 traffic, you can also enable IP Version 6 (IPv6) unicast routing and configure interfaces to forward IPv6 traffic if the switch or switch stack is running the IP Base or IP Services feature set.

Information About IP Routing

In some network environments, VLANs are associated with individual networks or subnetworks. In an IP network, each subnetwork is mapped to an individual VLAN. Configuring VLANs helps control the size of the broadcast domain and keeps local traffic local. However, network devices in different VLANs cannot communicate with one another without a Layer 3 device (router) to route traffic between the VLAN, referred to as inter-VLAN routing. You configure one or more routers to route traffic to the appropriate destination VLAN.

Figure 81: Routing Topology Example

This figure shows a basic routing topology. Switch A is in VLAN 10, and Switch B is in VLAN 20. The router



has an interface in each VLAN.

When Host A in VLAN 10 needs to communicate with Host B in VLAN 10, it sends a packet addressed to that host. Switch A forwards the packet directly to Host B, without sending it to the router.

When Host A sends a packet to Host C in VLAN 20, Switch A forwards the packet to the router, which receives the traffic on the VLAN 10 interface. The router checks the routing table, finds the correct outgoing interface, and forwards the packet on the VLAN 20 interface to Switch B. Switch B receives the packet and forwards it to Host C.

Types of Routing

Routers and Layer 3 switches can route packets in these ways:

- By using default routing
- By using preprogrammed static routes for the traffic
- By dynamically calculating routes by using a routing protocol

Default routing refers to sending traffic with a destination unknown to the router to a default outlet or destination.

Static unicast routing forwards packets from predetermined ports through a single path into and out of a network. Static routing is secure and uses little bandwidth, but does not automatically respond to changes in the network, such as link failures, and therefore, might result in unreachable destinations. As networks grow, static routing becomes a labor-intensive liability.

Switches running the LAN base feature set support 16 user-configured static routes, in addition to any default routes used for the management interface. The LAN base image supports static routing only on SVIs.

Dynamic routing protocols are used by routers to dynamically calculate the best route for forwarding traffic. There are two types of dynamic routing protocols:

- Routers using distance-vector protocols maintain routing tables with distance values of networked resources, and periodically pass these tables to their neighbors. Distance-vector protocols use one or a series of metrics for calculating the best routes. These protocols are easy to configure and use.

- Routers using link-state protocols maintain a complex database of network topology, based on the exchange of link-state advertisements (LSAs) between routers. LSAs are triggered by an event in the network, which speeds up the convergence time or time required to respond to these changes. Link-state protocols respond quickly to topology changes, but require greater bandwidth and more resources than distance-vector protocols.

Distance-vector protocols supported by the switch are Routing Information Protocol (RIP), which uses a single distance metric (cost) to determine the best path. The switch also supports the Open Shortest Path First (OSPF) link-state protocol, which adds some link-state routing features to traditional Interior Gateway Routing Protocol (IGRP) to improve efficiency.

IP Routing and Switch Stacks

A switch stack appears to the network as a single switch, regardless of which switch in the stack is connected to a routing peer.

The active switch performs these functions:

- It initializes and configures the routing protocols.
- It sends routing protocol messages and updates to other routers.
- It processes routing protocol messages and updates received from peer routers.
- It generates, maintains, and distributes the distributed Cisco Express Forwarding (dCEF) database to all stack members. The routes are programmed on all switches in the stack bases on this database.
- The MAC address of the active switch is used as the router MAC address for the whole stack, and all outside devices use this address to send IP packets to the stack.
- All IP packets that require software forwarding or processing go through the CPU of the active switch.

Stack members perform these functions:

- They act as routing standby switches, ready to take over in case they are elected as the new active switch if the active switch fails.
- They program the routes into hardware.

If a active switch fails, the stack detects that the active switch is down and elects one of the stack members to be the new active switch. During this period, except for a momentary interruption, the hardware continues to forward packets with no active protocols.

However, even though the switch stack maintains the hardware identification after a failure, the routing protocols on the router neighbors might flap during the brief interruption before the active switch restarts. Routing protocols such as OSPF and EIGRP need to recognize neighbor transitions.

Upon election, the new active switch performs these functions:

- It starts generating, receiving, and processing routing updates.
- It builds routing tables, generates the CEF database, and distributes it to stack members.
- It uses its MAC address as the router MAC address. To notify its network peers of the new MAC address, it periodically (every few seconds for 5 minutes) sends a gratuitous ARP reply with the new router MAC address.



Note If you configure the persistent MAC address feature on the stack and the active switch changes, the stack MAC address does not change for the configured time period. If the previous active switch rejoins the stack as a member switch during that time period, the stack MAC address remains the MAC address of the previous active switch.

- It attempts to determine the reachability of every proxy ARP entry by sending an ARP request to the proxy ARP IP address and receiving an ARP reply. For each reachable proxy ARP IP address, it generates a gratuitous ARP reply with the new router MAC address. This process is repeated for 5 minutes after a new active switch election.



Caution Partitioning of the switch stack into two or more stacks might lead to undesirable behavior in the network.

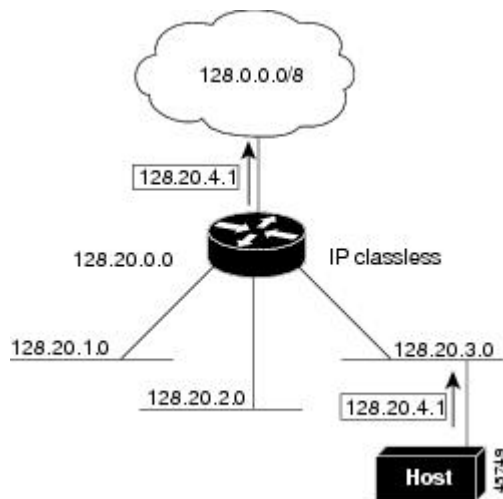
If the switch is reloaded, then all the ports on that switch go down and there is a loss of traffic for the interfaces involved in routing.

Classless Routing

By default, classless routing behavior is enabled on the Device when it is configured to route. With classless routing, if a router receives packets for a subnet of a network with no default route, the router forwards the packet to the best supernet route. A supernet consists of contiguous blocks of Class C address spaces used to simulate a single, larger address space and is designed to relieve the pressure on the rapidly depleting Class B address space.

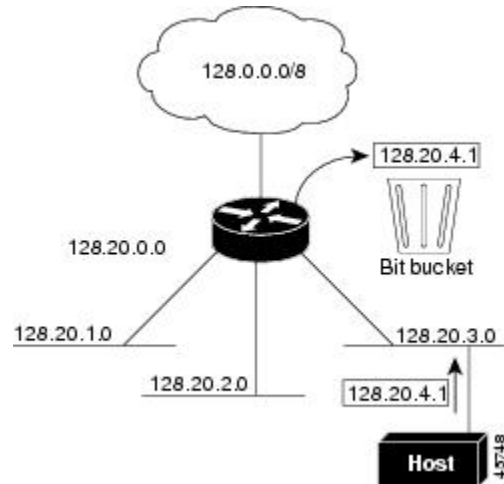
In the figure, classless routing is enabled. When the host sends a packet to 120.20.4.1, instead of discarding the packet, the router forwards it to the best supernet route. If you disable classless routing and a router receives packets destined for a subnet of a network with no network default route, the router discards the packet.

Figure 82: IP Classless Routing



In the figure, the router in network 128.20.0.0 is connected to subnets 128.20.1.0, 128.20.2.0, and 128.20.3.0. If the host sends a packet to 128.20.4.1, because there is no network default route, the router discards the packet.

Figure 83: No IP Classless Routing



To prevent the Device from forwarding packets destined for unrecognized subnets to the best supernet route possible, you can disable classless routing behavior.

Address Resolution

You can control interface-specific handling of IP by using address resolution. A device using IP can have both a local address or MAC address, which uniquely defines the device on its local segment or LAN, and a network address, which identifies the network to which the device belongs.



Note In a switch stack, network communication uses a single MAC address and the IP address of the stack.

The local address or MAC address is known as a data link address because it is contained in the data link layer (Layer 2) section of the packet header and is read by data link (Layer 2) devices. To communicate with a device on Ethernet, the software must learn the MAC address of the device. The process of learning the MAC address from an IP address is called *address resolution*. The process of learning the IP address from the MAC address is called *reverse address resolution*.

The Device can use these forms of address resolution:

- Address Resolution Protocol (ARP) is used to associate IP address with MAC addresses. Taking an IP address as input, ARP learns the associated MAC address and then stores the IP address/MAC address association in an ARP cache for rapid retrieval. Then the IP datagram is encapsulated in a link-layer frame and sent over the network. Encapsulation of IP datagrams and ARP requests or replies on IEEE 802 networks other than Ethernet is specified by the Subnetwork Access Protocol (SNAP).
- Proxy ARP helps hosts with no routing tables learn the MAC addresses of hosts on other networks or subnets. If the Device (router) receives an ARP request for a host that is not on the same interface as the ARP request sender, and if the router has all of its routes to the host through other interfaces, it generates

a proxy ARP packet giving its own local data link address. The host that sent the ARP request then sends its packets to the router, which forwards them to the intended host.

The Device also uses the Reverse Address Resolution Protocol (RARP), which functions the same as ARP does, except that the RARP packets request an IP address instead of a local MAC address. Using RARP requires a RARP server on the same network segment as the router interface. Use the **ip rarp-server address** interface configuration command to identify the server.

For more information on RARP, see the *Cisco IOS Configuration Fundamentals Configuration Guide*

Proxy ARP

Proxy ARP, the most common method for learning about other routes, enables an Ethernet host with no routing information to communicate with hosts on other networks or subnets. The host assumes that all hosts are on the same local Ethernet and that they can use ARP to learn their MAC addresses. If a Device receives an ARP request for a host that is not on the same network as the sender, the Device evaluates whether it has the best route to that host. If it does, it sends an ARP reply packet with its own Ethernet MAC address, and the host that sent the request sends the packet to the Device, which forwards it to the intended host. Proxy ARP treats all networks as if they are local, and performs ARP requests for every IP address.

ICMP Router Discovery Protocol

Router discovery allows the Device to dynamically learn about routes to other networks using ICMP router discovery protocol (IRDP). IRDP allows hosts to locate routers. When operating as a client, the Device generates router discovery packets. When operating as a host, the Device receives router discovery packets. The Device can also listen to Routing Information Protocol (RIP) routing updates and use this information to infer locations of routers. The Device does not actually store the routing tables sent by routing devices; it merely keeps track of which systems are sending the data. The advantage of using IRDP is that it allows each router to specify both a priority and the time after which a device is assumed to be down if no further packets are received.

Each device discovered becomes a candidate for the default router, and a new highest-priority router is selected when a higher priority router is discovered, when the current default router is declared down, or when a TCP connection is about to time out because of excessive retransmissions.

UDP Broadcast Packets and Protocols

User Datagram Protocol (UDP) is an IP host-to-host layer protocol, as is TCP. UDP provides a low-overhead, connectionless session between two end systems and does not provide for acknowledgment of received datagrams. Network hosts occasionally use UDP broadcasts to find address, configuration, and name information. If such a host is on a network segment that does not include a server, UDP broadcasts are normally not forwarded. You can remedy this situation by configuring an interface on a router to forward certain classes of broadcasts to a helper address. You can use more than one helper address per interface.

You can specify a UDP destination port to control which UDP services are forwarded. You can specify multiple UDP protocols. You can also specify the Network Disk (ND) protocol, which is used by older diskless Sun workstations and the network security protocol SDNS.

By default, both UDP and ND forwarding are enabled if a helper address has been defined for an interface. The description for the **ip forward-protocol** interface configuration command in the *Cisco IOS IP Command Reference, Volume 1 of 3: Addressing and Services* lists the ports that are forwarded by default if you do not specify any UDP ports.

Broadcast Packet Handling

After configuring an IP interface address, you can enable routing and configure one or more routing protocols, or you can configure the way the Device responds to network broadcasts. A broadcast is a data packet destined for all hosts on a physical network. The Device supports two kinds of broadcasting:

- A directed broadcast packet is sent to a specific network or series of networks. A directed broadcast address includes the network or subnet fields.
- A flooded broadcast packet is sent to every network.



Note You can also limit broadcast, unicast, and multicast traffic on Layer 2 interfaces by using the **storm-control** interface configuration command to set traffic suppression levels.

Routers provide some protection from broadcast storms by limiting their extent to the local cable. Bridges (including intelligent bridges), because they are Layer 2 devices, forward broadcasts to all network segments, thus propagating broadcast storms. The best solution to the broadcast storm problem is to use a single broadcast address scheme on a network. In most modern IP implementations, you can set the address to be used as the broadcast address. Many implementations, including the one in the Device, support several addressing schemes for forwarding broadcast messages.

IP Broadcast Flooding

You can allow IP broadcasts to be flooded throughout your internetwork in a controlled fashion by using the database created by the bridging STP. Using this feature also prevents loops. To support this capability, bridging must be configured on each interface that is to participate in the flooding. If bridging is not configured on an interface, it still can receive broadcasts. However, the interface never forwards broadcasts it receives, and the router never uses that interface to send broadcasts received on a different interface.

Packets that are forwarded to a single network address using the IP helper-address mechanism can be flooded. Only one copy of the packet is sent on each network segment.

To be considered for flooding, packets must meet these criteria. (Note that these are the same conditions used to consider packet forwarding using IP helper addresses.)

- The packet must be a MAC-level broadcast.
- The packet must be an IP-level broadcast.
- The packet must be a TFTP, DNS, Time, NetBIOS, ND, or BOOTP packet, or a UDP specified by the **ip forward-protocol udp** global configuration command.
- The time-to-live (TTL) value of the packet must be at least two.

A flooded UDP datagram is given the destination address specified with the **ip broadcast-address** interface configuration command on the output interface. The destination address can be set to any address. Thus, the destination address might change as the datagram propagates through the network. The source address is never changed. The TTL value is decremented.

When a flooded UDP datagram is sent out an interface (and the destination address possibly changed), the datagram is handed to the normal IP output routines and is, therefore, subject to access lists, if they are present on the output interface.

In the Device, the majority of packets are forwarded in hardware; most packets do not go through the Device CPU. For those packets that do go to the CPU, you can speed up spanning tree-based UDP flooding by a factor of about four to five times by using turbo-flooding. This feature is supported over Ethernet interfaces configured for ARP encapsulation.

How to Configure IP Routing

By default, IP routing is disabled on the Device, and you must enable it before routing can take place. For detailed IP routing configuration information, see the *Cisco IOS IP Configuration Guide*.

In the following procedures, the specified interface must be one of these Layer 3 interfaces:

- A routed port: a physical port configured as a Layer 3 port by using the **no switchport** interface configuration command.
- A switch virtual interface (SVI): a VLAN interface created by using the **interface vlan** *vlan_id* global configuration command and by default a Layer 3 interface.
- An EtherChannel port channel in Layer 3 mode: a port-channel logical interface created by using the **interface port-channel** *port-channel-number* global configuration command and binding the Ethernet interface into the channel group. For more information, see the “Configuring Layer 3 EtherChannels” chapter in the Layer 2 Configuration Guide.



Note The switch does not support tunnel interfaces for unicast routed traffic.

All Layer 3 interfaces on which routing will occur must have IP addresses assigned to them.



Note A Layer 3 switch can have an IP address assigned to each routed port and SVI.

Configuring routing consists of several main procedures:

- To support VLAN interfaces, create and configure VLANs on the Device or switch stack, and assign VLAN membership to Layer 2 interfaces. For more information, see the “Configuring VLANs” chapter in the VLAN Configuration Guide.
- Configure Layer 3 interfaces.
- Enable IP routing on the switch.
- Assign IP addresses to the Layer 3 interfaces.
- Enable selected routing protocols on the switch.
- Configure routing protocol parameters (optional).

How to Configure IP Addressing

A required task for configuring IP routing is to assign IP addresses to Layer 3 network interfaces to enable the interfaces and allow communication with the hosts on those interfaces that use IP. The following sections describe how to configure various IP addressing features. Assigning IP addresses to the interface is required; the other procedures are optional.

- Default Addressing Configuration
- Assigning IP Addresses to Network Interfaces
- Configuring Address Resolution Methods
- Routing Assistance When IP Routing is Disabled
- Configuring Broadcast Packet Handling
- Monitoring and Maintaining IP Addressing

Default IP Addressing Configuration

Table 91: Default Addressing Configuration

Feature	Default Setting
IP address	None defined.
ARP	No permanent entries in the Address Resolution Protocol (ARP) cache. Encapsulation: Standard Ethernet-style ARP. Timeout: 14400 seconds (4 hours).
IP broadcast address	255.255.255.255 (all ones).
IP classless routing	Enabled.
IP default gateway	Disabled.
IP directed broadcast	Disabled (all IP directed broadcasts are dropped).
IP domain	Domain list: No domain names defined. Domain lookup: Enabled. Domain name: Enabled.

Feature	Default Setting
IP forward-protocol	If a helper address is defined or User Datagram Protocol (UDP) flooding is configured, UDP forwarding is enabled on default ports. Any-local-broadcast: Disabled. Spanning Tree Protocol (STP): Disabled. Turbo-flood: Disabled.
IP helper address	Disabled.
IP host	Disabled.
IRDP	Disabled. Defaults when enabled: <ul style="list-style-type: none"> • Broadcast IRDP advertisements. • Maximum interval between advertisements: 600 seconds. • Minimum interval between advertisements: 0.75 times max interval • Preference: 0.
IP proxy ARP	Enabled.
IP routing	Disabled.
IP subnet-zero	Disabled.

Assigning IP Addresses to Network Interfaces

An IP address identifies a location to which IP packets can be sent. Some IP addresses are reserved for special uses and cannot be used for host, subnet, or network addresses. RFC 1166, “Internet Numbers,” contains the official description of IP addresses.

An interface can have one primary IP address. A mask identifies the bits that denote the network number in an IP address. When you use the mask to subnet a network, the mask is referred to as a subnet mask. To receive an assigned network number, contact your Internet service provider.

Procedure

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.

	Command or Action	Purpose
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	interface <i>interface-id</i> Example: Device(config)# interface gigabitethernet 1/0/1	Enters interface configuration mode, and specifies the Layer 3 interface to configure.
Step 4	no switchport Example: Device(config-if)# no switchport	Removes the interface from Layer 2 configuration mode (if it is a physical interface).
Step 5	ip address <i>ip-address subnet-mask</i> Example: Device(config-if)# ip address 10.1.5.1 255.255.255.0	Configures the IP address and IP subnet mask.
Step 6	no shutdown Example: Device(config-if)# no shutdown	Enables the physical interface.
Step 7	end Example: Device(config)# end	Returns to privileged EXEC mode.
Step 8	show ip route Example: Device# show ip route	Verifies your entries.
Step 9	show ip interface [<i>interface-id</i>] Example: Device# show ip interface gigabitethernet 1/0/1	Verifies your entries.
Step 10	show running-config Example: Device# show running-config	Verifies your entries.

	Command or Action	Purpose
Step 11	copy running-config startup-config Example: Device# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Using Subnet Zero

Subnetting with a subnet address of zero is strongly discouraged because of the problems that can arise if a network and a subnet have the same addresses. For example, if network 131.108.0.0 is subnetted as 255.255.255.0, subnet zero would be written as 131.108.0.0, which is the same as the network address.

You can use the all ones subnet (131.108.255.0) and even though it is discouraged, you can enable the use of subnet zero if you need the entire subnet space for your IP address.

Use the **no ip subnet-zero** global configuration command to restore the default and disable the use of subnet zero.

Procedure

	Command or Action	Purpose
Step 1	enable Example: Device> <code>enable</code>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode.
Step 3	ip subnet-zero Example: Device(config)# <code>ip subnet-zero</code>	Enables the use of subnet zero for interface addresses and routing updates.
Step 4	end Example: Device(config)# <code>end</code>	Returns to privileged EXEC mode.
Step 5	show running-config Example: Device# <code>show running-config</code>	Verifies your entries.

	Command or Action	Purpose
Step 6	copy running-config startup-config Example: Device# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Disabling Classless Routing

To prevent the Device from forwarding packets destined for unrecognized subnets to the best supernet route possible, you can disable classless routing behavior.

Procedure

	Command or Action	Purpose
Step 1	enable Example: Device> <code>enable</code>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode.
Step 3	no ip classless Example: Device(config)# <code>no ip classless</code>	Disables classless routing behavior.
Step 4	end Example: Device(config)# <code>end</code>	Returns to privileged EXEC mode.
Step 5	show running-config Example: Device# <code>show running-config</code>	Verifies your entries.
Step 6	copy running-config startup-config Example:	(Optional) Saves your entries in the configuration file.

	Command or Action	Purpose
	Device# <code>copy running-config startup-config</code>	

Configuring Address Resolution Methods

You can perform the following tasks to configure address resolution.

Defining a Static ARP Cache

ARP and other address resolution protocols provide dynamic mapping between IP addresses and MAC addresses. Because most hosts support dynamic address resolution, you usually do not need to specify static ARP cache entries. If you must define a static ARP cache entry, you can do so globally, which installs a permanent entry in the ARP cache that the Device uses to translate IP addresses into MAC addresses. Optionally, you can also specify that the Device respond to ARP requests as if it were the owner of the specified IP address. If you do not want the ARP entry to be permanent, you can specify a timeout period for the ARP entry.

Procedure

	Command or Action	Purpose
Step 1	enable Example: Device> <code>enable</code>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode.
Step 3	arp ip-address hardware-address type Example: Device(config)# <code>ip 10.1.5.1 c2f3.220a.12f4 arpa</code>	Associates an IP address with a MAC (hardware) address in the ARP cache, and specifies encapsulation type as one of these: <ul style="list-style-type: none"> • arpa—ARP encapsulation for Ethernet interfaces • snap—Subnetwork Address Protocol encapsulation for Token Ring and FDDI interfaces • sap—HP's ARP type
Step 4	arp ip-address hardware-address type [alias] Example: Device(config)# <code>ip 10.1.5.3 d7f3.220d.12f5 arpa alias</code>	(Optional) Specifies that the switch respond to ARP requests as if it were the owner of the specified IP address.

	Command or Action	Purpose
Step 5	interface <i>interface-id</i> Example: Device(config)# interface gigabitethernet 1/0/1	Enters interface configuration mode, and specifies the interface to configure.
Step 6	arp <i>timeout seconds</i> Example: Device(config-if)# arp 20000	(Optional) Sets the length of time an ARP cache entry will stay in the cache. The default is 14400 seconds (4 hours). The range is 0 to 2147483 seconds.
Step 7	end Example: Device(config)# end	Returns to privileged EXEC mode.
Step 8	show interfaces [<i>interface-id</i>] Example: Device# show interfaces gigabitethernet 1/0/1	Verifies the type of ARP and the timeout value used on all interfaces or a specific interface.
Step 9	show arp Example: Device# show arp	Views the contents of the ARP cache.
Step 10	show ip arp Example: Device# show ip arp	Views the contents of the ARP cache.
Step 11	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Setting ARP Encapsulation

By default, Ethernet ARP encapsulation (represented by the **arpa** keyword) is enabled on an IP interface. You can change the encapsulation methods to SNAP if required by your network.

To disable an encapsulation type, use the **no arp arpa** or **no arp snap** interface configuration command.

Procedure

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none">• Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	interface <i>interface-id</i> Example: Device(config)# interface gigabitethernet 1/0/2	Enters interface configuration mode, and specifies the Layer 3 interface to configure.
Step 4	arp {arpa snap} Example: Device(config-if)# arp arpa	Specifies the ARP encapsulation method: <ul style="list-style-type: none">• arpa—Address Resolution Protocol• snap—Subnetwork Address Protocol
Step 5	end Example: Device(config)# end	Returns to privileged EXEC mode.
Step 6	show interfaces [<i>interface-id</i>] Example: Device# show interfaces	Verifies ARP encapsulation configuration on all interfaces or the specified interface.
Step 7	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Enabling Proxy ARP

By default, the Device uses proxy ARP to help hosts learn MAC addresses of hosts on other networks or subnets.

Procedure

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	interface <i>interface-id</i> Example: Device(config)# interface gigabitethernet 1/0/2	Enters interface configuration mode, and specifies the Layer 3 interface to configure.
Step 4	ip proxy-arp Example: Device(config-if)# ip proxy-arp	Enables proxy ARP on the interface.
Step 5	end Example: Device(config)# end	Returns to privileged EXEC mode.
Step 6	show ip interface [<i>interface-id</i>] Example: Device# show ip interface gigabitethernet 1/0/2	Verifies the configuration on the interface or all interfaces.
Step 7	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Routing Assistance When IP Routing is Disabled

These mechanisms allow the Device to learn about routes to other networks when it does not have IP routing enabled:

- Proxy ARP
- Default Gateway

- ICMP Router Discovery Protocol (IRDP)

Proxy ARP

Proxy ARP is enabled by default. To enable it after it has been disabled, see the “Enabling Proxy ARP” section. Proxy ARP works as long as other routers support it.

Default Gateway

Another method for locating routes is to define a default router or default gateway. All non-local packets are sent to this router, which either routes them appropriately or sends an IP Control Message Protocol (ICMP) redirect message back, defining which local router the host should use. The Device caches the redirect messages and forwards each packet as efficiently as possible. A limitation of this method is that there is no means of detecting when the default router has gone down or is unavailable.

Procedure

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	ip default-gateway <i>ip-address</i> Example: Device(config)# ip default gateway 10.1.5.1	Sets up a default gateway (router).
Step 4	end Example: Device(config)# end	Returns to privileged EXEC mode.
Step 5	show ip redirects Example: Device# show ip redirects	Displays the address of the default gateway router to verify the setting.
Step 6	copy running-config startup-config Example:	(Optional) Saves your entries in the configuration file.

	Command or Action	Purpose
	Device# <code>copy running-config startup-config</code>	

ICMP Router Discovery Protocol (IRDP)

The only required task for IRDP routing on an interface is to enable IRDP processing on that interface. When enabled, the default parameters apply.

You can optionally change any of these parameters. If you change the **maxadvertinterval** value, the **holdtime** and **minadvertinterval** values also change, so it is important to first change the **maxadvertinterval** value, before manually changing either the **holdtime** or **minadvertinterval** values.

Procedure

	Command or Action	Purpose
Step 1	enable Example: Device> <code>enable</code>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode.
Step 3	interface <i>interface-id</i> Example: Device(config)# <code>interface gigabitethernet 1/0/1</code>	Enters interface configuration mode, and specifies the Layer 3 interface to configure.
Step 4	ip irdp Example: Device(config-if)# <code>ip irdp</code>	Enables IRDP processing on the interface.
Step 5	ip irdp multicast Example: Device(config-if)# <code>ip irdp multicast</code>	(Optional) Sends IRDP advertisements to the multicast address (224.0.0.1) instead of IP broadcasts. Note This command allows for compatibility with Sun Microsystems Solaris, which requires IRDP packets to be sent out as multicasts. Many implementations cannot receive these multicasts; ensure end-host ability before using this command.

	Command or Action	Purpose
Step 6	ip irdp holdtime <i>seconds</i> Example: Device(config-if)# ip irdp holdtime 1000	(Optional) Sets the IRDP period for which advertisements are valid. The default is three times the maxadvertinterval value. It must be greater than maxadvertinterval and cannot be greater than 9000 seconds. If you change the maxadvertinterval value, this value also changes.
Step 7	ip irdp maxadvertinterval <i>seconds</i> Example: Device(config-if)# ip irdp maxadvertinterval 650	(Optional) Sets the IRDP maximum interval between advertisements. The default is 600 seconds.
Step 8	ip irdp minadvertinterval <i>seconds</i> Example: Device(config-if)# ip irdp minadvertinterval 500	(Optional) Sets the IRDP minimum interval between advertisements. The default is 0.75 times the maxadvertinterval . If you change the maxadvertinterval , this value changes to the new default (0.75 of maxadvertinterval).
Step 9	ip irdp preference <i>number</i> Example: Device(config-if)# ip irdp preference 2	(Optional) Sets a device IRDP preference level. The allowed range is -231 to 231. The default is 0. A higher value increases the router preference level.
Step 10	ip irdp address <i>address [number]</i> Example: Device(config-if)# ip irdp address 10.1.10.10	(Optional) Specifies an IRDP address and preference to proxy-advertise.
Step 11	end Example: Device(config)# end	Returns to privileged EXEC mode.
Step 12	show ip irdp Example: Device# show ip irdp	Verifies settings by displaying IRDP values.
Step 13	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Configuring Broadcast Packet Handling

Perform the tasks in these sections to enable these schemes:

- Enabling Directed Broadcast-to-Physical Broadcast Translation
- Forwarding UDP Broadcast Packets and Protocols
- Establishing an IP Broadcast Address
- Flooding IP Broadcasts

Enabling Directed Broadcast-to-Physical Broadcast Translation

By default, IP directed broadcasts are dropped; they are not forwarded. Dropping IP-directed broadcasts makes routers less susceptible to denial-of-service attacks.

You can enable forwarding of IP-directed broadcasts on an interface where the broadcast becomes a physical (MAC-layer) broadcast. Only those protocols configured by using the **ip forward-protocol** global configuration command are forwarded.

You can specify an access list to control which broadcasts are forwarded. When an access list is specified, only those IP packets permitted by the access list are eligible to be translated from directed broadcasts to physical broadcasts. For more information on access lists, see the "Information about Network Security with ACLs" section in the Security Configuration Guide.

Procedure

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	interface <i>interface-id</i> Example: Device(config)# interface gigabitethernet 1/0/2	Enters interface configuration mode, and specifies the interface to configure.
Step 4	ip directed-broadcast [<i>access-list-number</i>] Example: Device(config-if)# ip directed-broadcast 103	Enables directed broadcast-to-physical broadcast translation on the interface. You can include an access list to control which broadcasts are forwarded. When an access list, only IP packets permitted by the access list can be translated.
Step 5	exit Example: Device(config-if)# exit	Returns to global configuration mode.

	Command or Action	Purpose
Step 6	ip forward-protocol {udp [port] nd sdns} Example: <pre>Device(config)# ip forward-protocol nd</pre>	Specifies which protocols and ports the router forwards when forwarding broadcast packets. <ul style="list-style-type: none"> • udp—Forward UPD datagrams. port: (Optional) Destination port that controls which UDP services are forwarded. • nd—Forward ND datagrams. • sdns—Forward SDNS datagrams
Step 7	end Example: <pre>Device(config)# end</pre>	Returns to privileged EXEC mode.
Step 8	show ip interface [interface-id] Example: <pre>Device# show ip interface</pre>	Verifies the configuration on the interface or all interfaces
Step 9	show running-config Example: <pre>Device# show running-config</pre>	Verifies your entries.
Step 10	copy running-config startup-config Example: <pre>Device# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Forwarding UDP Broadcast Packets and Protocols

If you do not specify any UDP ports when you configure the forwarding of UDP broadcasts, you are configuring the router to act as a BOOTP forwarding agent. BOOTP packets carry DHCP information.

Procedure

	Command or Action	Purpose
Step 1	enable Example: <pre>Device> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.

	Command or Action	Purpose
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	interface <i>interface-id</i> Example: Device(config)# interface gigabitethernet 1/0/1	Enters interface configuration mode, and specifies the Layer 3 interface to configure.
Step 4	ip helper-address <i>address</i> Example: Device(config-if)# ip helper address 10.1.10.1	Enables forwarding and specifies the destination address for forwarding UDP broadcast packets, including BOOTP.
Step 5	exit Example: Device(config-if)# exit	Returns to global configuration mode.
Step 6	ip forward-protocol {udp [<i>port</i>] nd sdns} Example: Device(config)# ip forward-protocol sdns	Specifies which protocols the router forwards when forwarding broadcast packets.
Step 7	end Example: Device(config)# end	Returns to privileged EXEC mode.
Step 8	show ip interface [<i>interface-id</i>] Example: Device# show ip interface gigabitethernet 1/0/1	Verifies the configuration on the interface or all interfaces.
Step 9	show running-config Example: Device# show running-config	Verifies your entries.
Step 10	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Establishing an IP Broadcast Address

The most popular IP broadcast address (and the default) is an address consisting of all ones (255.255.255.255). However, the Device can be configured to generate any form of IP broadcast address.

Procedure

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	interface <i>interface-id</i> Example: Device(config)# interface gigabitethernet 1/0/1	Enters interface configuration mode, and specifies the interface to configure.
Step 4	ip broadcast-address <i>ip-address</i> Example: Device(config-if)# ip broadcast-address 128.1.255.255	Enters a broadcast address different from the default, for example 128.1.255.255.
Step 5	end Example: Device(config)# end	Returns to privileged EXEC mode.
Step 6	show ip interface [<i>interface-id</i>] Example: Device# show ip interface	Verifies the broadcast address on the interface or all interfaces.
Step 7	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Flooding IP Broadcasts

Procedure

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	ip forward-protocol spanning-tree Example: Device(config)# ip forward-protocol spanning-tree	Uses the bridging spanning-tree database to flood UDP datagrams.
Step 4	end Example: Device(config)# end	Returns to privileged EXEC mode.
Step 5	show running-config Example: Device# show running-config	Verifies your entries.
Step 6	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.
Step 7	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 8	ip forward-protocol turbo-flood Example:	Uses the spanning-tree database to speed up flooding of UDP datagrams.

	Command or Action	Purpose
	<code>Device(config)# ip forward-protocol turbo-flood</code>	
Step 9	end Example: <code>Device(config)# end</code>	Returns to privileged EXEC mode.
Step 10	show running-config Example: <code>Device# show running-config</code>	Verifies your entries.
Step 11	copy running-config startup-config Example: <code>Device# copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Monitoring and Maintaining IP Addressing

When the contents of a particular cache, table, or database have become or are suspected to be invalid, you can remove all its contents by using the **clear** privileged EXEC commands. The Table lists the commands for clearing contents.

Table 92: Commands to Clear Caches, Tables, and Databases

clear arp-cache	Clears the IP ARP cache and the fast-switching cache.
clear host { <i>name</i> *}	Removes one or all entries from the hostname and the address cache.
clear ip route { network [<i>mask</i>] *}	Removes one or more routes from the IP routing table.

You can display specific statistics, such as the contents of IP routing tables, caches, and databases; the reachability of nodes; and the routing path that packets are taking through the network. The Table lists the privileged EXEC commands for displaying IP statistics.

Table 93: Commands to Display Caches, Tables, and Databases

show arp	Displays the entries in the ARP table.
show hosts	Displays the default domain name, style of lookup service, name server hosts, and the cached list of hostnames and addresses.
show ip aliases	Displays IP addresses mapped to TCP ports (aliases).

show ip arp	Displays the IP ARP cache.
show ip interface [<i>interface-id</i>]	Displays the IP status of interfaces.
show ip irdp	Displays IRDP values.
show ip masks <i>address</i>	Displays the masks used for network addresses and the number of subnets using each mask.
show ip redirects	Displays the address of a default gateway.
show ip route [<i>address</i> [<i>mask</i>]] [<i>protocol</i>]	Displays the current state of the routing table.
show ip route summary	Displays the current state of the routing table in summary form.

How to Configure IP Unicast Routing

Enabling IP Unicast Routing

By default, the Device is in Layer 2 switching mode and IP routing is disabled. To use the Layer 3 capabilities of the Device, you must enable IP routing.

Procedure

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	ip routing Example: Device(config)# ip routing	Enables IP routing.
Step 4	end Example: Device(config)# end	Returns to privileged EXEC mode.

	Command or Action	Purpose
Step 5	show running-config Example: Device# <code>show running-config</code>	Verifies your entries.
Step 6	copy running-config startup-config Example: Device# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Example of Enabling IP Routing

This example shows how to enable IP routing :

```
Device# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Device(config)# ip routing

Device(config-router)# end
```

What to Do Next

You can now set up parameters for the selected routing protocols as described in these sections:

- RIP
- OSPF,
- EIGRP
- Unicast Reverse Path Forwarding
- Protocol-Independent Features (optional)

Information About RIP

The Routing Information Protocol (RIP) is an interior gateway protocol (IGP) created for use in small, homogeneous networks. It is a distance-vector routing protocol that uses broadcast User Datagram Protocol (UDP) data packets to exchange routing information. The protocol is documented in RFC 1058. You can find detailed information about RIP in *IP Routing Fundamentals*, published by Cisco Press.



Note RIP is supported in the IP Lite Network Essentials feature set.

Using RIP, the Device sends routing information updates (advertisements) every 30 seconds. If a router does not receive an update from another router for 180 seconds or more, it marks the routes served by that router as unusable. If there is still no update after 240 seconds, the router removes all routing table entries for the non-updating router.

RIP uses hop counts to rate the value of different routes. The hop count is the number of routers that can be traversed in a route. A directly connected network has a hop count of zero; a network with a hop count of 16 is unreachable. This small range (0 to 15) makes RIP unsuitable for large networks.

If the router has a default network path, RIP advertises a route that links the router to the pseudonetwork 0.0.0.0. The 0.0.0.0 network does not exist; it is treated by RIP as a network to implement the default routing feature. The Device advertises the default network if a default was learned by RIP or if the router has a gateway of last resort and RIP is configured with a default metric. RIP sends updates to the interfaces in specified networks. If an interface's network is not specified, it is not advertised in any RIP update.

Summary Addresses and Split Horizon

Routers connected to broadcast-type IP networks and using distance-vector routing protocols normally use the split-horizon mechanism to reduce the possibility of routing loops. Split horizon blocks information about routes from being advertised by a router on any interface from which that information originated. This feature usually optimizes communication among multiple routers, especially when links are broken.

How to Configure RIP

Default RIP Configuration

Table 94: Default RIP Configuration

Feature	Default Setting
Auto summary	Enabled.
Default-information originate	Disabled.
Default metric	Built-in; automatic metric translations.
IP RIP authentication key-chain	No authentication. Authentication mode: clear text.
IP RIP triggered	Disabled
IP split horizon	Varies with media.
Neighbor	None defined.
Network	None specified.
Offset list	Disabled.
Output delay	0 milliseconds.

Feature	Default Setting
Timers basic	<ul style="list-style-type: none"> • Update: 30 seconds. • Invalid: 180 seconds. • Hold-down: 180 seconds. • Flush: 240 seconds.
Validate-update-source	Enabled.
Version	Receives RIP Version 1 and 2 packets; sends Version 1 packets.

Configuring Basic RIP Parameters

To configure RIP, you enable RIP routing for a network and optionally configure other parameters. On the Device, RIP configuration commands are ignored until you configure the network number.

Procedure

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	ip routing Example: Device(config)# ip routing	Enables IP routing. (Required only if IP routing is disabled.)
Step 4	router rip Example: Device(config)# router rip	Enables a RIP routing process, and enter router configuration mode.
Step 5	network <i>network number</i> Example: Device(config-router)# network 12.0.0.0	Associates a network with a RIP routing process. You can specify multiple network commands. RIP routing updates are sent and received through interfaces only on these networks.

	Command or Action	Purpose
		Note You must configure a network number for the RIP commands to take effect.
Step 6	neighbor <i>ip-address</i> Example: Device(config-router)# neighbor 10.2.5.1	(Optional) Defines a neighboring router with which to exchange routing information. This step allows routing updates from RIP (normally a broadcast protocol) to reach nonbroadcast networks.
Step 7	offset-list [<i>access-list number name</i>] { in out } <i>offset</i> [<i>type number</i>] Example: Device(config-router)# offset-list 103 in 10	(Optional) Applies an offset list to routing metrics to increase incoming and outgoing metrics to routes learned through RIP. You can limit the offset list with an access list or an interface.
Step 8	timers basic <i>update invalid holddown flush</i> Example: Device(config-router)# timers basic 45 360 400 300	(Optional) Adjusts routing protocol timers. Valid ranges for all timers are 0 to 4294967295 seconds. <ul style="list-style-type: none"> • <i>update</i>—The time between sending routing updates. The default is 30 seconds. • <i>invalid</i>—The timer after which a route is declared invalid. The default is 180 seconds. • <i>holddown</i>—The time before a route is removed from the routing table. The default is 180 seconds. • <i>flush</i>—The amount of time for which routing updates are postponed. The default is 240 seconds.
Step 9	version { 1 2 } Example: Device(config-router)# version 2	(Optional) Configures the switch to receive and send only RIP Version 1 or RIP Version 2 packets. By default, the switch receives Version 1 and 2 but sends only Version 1. You can also use the interface commands ip rip {send receive} version 1 2 1 2 to control what versions are used for sending and receiving on interfaces.
Step 10	no auto-summary Example: Device(config-router)# no auto-summary	(Optional) Disables automatic summarization. By default, the switch summarizes subprefixes when crossing classful network boundaries. Disable summarization (RIP Version 2 only) to advertise subnet and host routing information to classful network boundaries.
Step 11	output-delay <i>delay</i> Example: Device(config-router)# output-delay 8	(Optional) Adds interpacket delay for RIP updates sent. By default, packets in a multiple-packet RIP update have no delay added between packets. If you are sending packets to a lower-speed device, you can add an interpacket delay in the range of 8 to 50 milliseconds.
Step 12	end Example:	Returns to privileged EXEC mode.

	Command or Action	Purpose
	Device(config-router)# end	
Step 13	show ip protocols Example: Device# show ip protocols	Verifies your entries.
Step 14	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Configuring RIP Authentication

RIP Version 1 does not support authentication. If you are sending and receiving RIP Version 2 packets, you can enable RIP authentication on an interface. The key chain specifies the set of keys that can be used on the interface. If a key chain is not configured, no authentication is performed, not even the default.

The Device supports two modes of authentication on interfaces for which RIP authentication is enabled: plain text and MD5. The default is plain text.

Procedure

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	interface <i>interface-id</i> Example: Device(config)# interface gigabitethernet 1/0/1	Enters interface configuration mode, and specifies the interface to configure.
Step 4	ip rip authentication key-chain <i>name-of-chain</i> Example: Device(config-if)# ip rip authentication key-chain trees	Enables RIP authentication.

	Command or Action	Purpose
Step 5	ip rip authentication mode {text md5} Example: Device(config-if)# ip rip authentication mode md5	Configures the interface to use plain text authentication (the default) or MD5 digest authentication.
Step 6	end Example: Device(config)# end	Returns to privileged EXEC mode.
Step 7	show running-config Example: Device# show running-config	Verifies your entries.
Step 8	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Configuring Summary Addresses and Split Horizon



Note In general, disabling split horizon is not recommended unless you are certain that your application requires it to properly advertise routes.

If you want to configure an interface running RIP to advertise a summarized local IP address pool on a network access server for dial-up clients, use the **ip summary-address rip** interface configuration command.



Note If split horizon is enabled, neither autosummary nor interface IP summary addresses are advertised.

Procedure

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.

	Command or Action	Purpose
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	interface <i>interface-id</i> Example: Device(config)# interface gigabitethernet 1/0/1	Enters interface configuration mode, and specifies the Layer 3 interface to configure.
Step 4	ip address <i>ip-address subnet-mask</i> Example: Device(config-if)# ip address 10.1.1.10 255.255.255.0	Configures the IP address and IP subnet.
Step 5	ip summary-address rip <i>ip-address ip-network mask</i> Example: Device(config-if)# ip summary-address rip 10.1.1.30 255.255.255.0	Configures the IP address to be summarized and the IP network mask.
Step 6	no ip split-horizon Example: Device(config-if)# no ip split-horizon	Disables split horizon on the interface.
Step 7	end Example: Device(config)# end	Returns to privileged EXEC mode.
Step 8	show ip interface <i>interface-id</i> Example: Device# show ip interface gigabitethernet 1/0/1	Verifies your entries.
Step 9	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Configuring Split Horizon

Routers connected to broadcast-type IP networks and using distance-vector routing protocols normally use the split-horizon mechanism to reduce the possibility of routing loops. Split horizon blocks information about routes from being advertised by a router on any interface from which that information originated. This feature can optimize communication among multiple routers, especially when links are broken.



Note In general, we do not recommend disabling split horizon unless you are certain that your application requires it to properly advertise routes.

Procedure

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	interface <i>interface-id</i> Example: Device(config)# interface gigabitethernet 1/0/1	Enters interface configuration mode, and specifies the interface to configure.
Step 4	ip address <i>ip-address subnet-mask</i> Example: Device(config-if)# ip address 10.1.1.10 255.255.255.0	Configures the IP address and IP subnet.
Step 5	no ip split-horizon Example: Device(config-if)# no ip split-horizon	Disables split horizon on the interface.
Step 6	end Example: Device(config)# end	Returns to privileged EXEC mode.

	Command or Action	Purpose
Step 7	show ip interface <i>interface-id</i> Example: Device# show ip interface gigabitethernet 1/0/1	Verifies your entries.
Step 8	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Configuration Example for Summary Addresses and Split Horizon

In this example, the major net is 10.0.0.0. The summary address 10.2.0.0 overrides the autosummary address of 10.0.0.0 so that 10.2.0.0 is advertised out interface Gigabit Ethernet port 2, and 10.0.0.0 is not advertised. In the example, if the interface is still in Layer 2 mode (the default), you must enter a **no switchport** interface configuration command before entering the **ip address** interface configuration command.



Note If split horizon is enabled, neither autosummary nor interface summary addresses (those configured with the **ip summary-address rip** router configuration command) are advertised.

```
Device(config)# router rip
Device(config-router)# interface gigabitethernet1/0/2
Device(config-if)# ip address 10.1.5.1 255.255.255.0
Device(config-if)# ip summary-address rip 10.2.0.0 255.255.0.0
Device(config-if)# no ip split-horizon
Device(config-if)# exit
Device(config)# router rip
Device(config-router)# network 10.0.0.0
Device(config-router)# neighbor 2.2.2.2 peer-group mygroup
Device(config-router)# end
```

Information About OSPF

OSPF is an Interior Gateway Protocol (IGP) designed expressly for IP networks, supporting IP subnetting and tagging of externally derived routing information. OSPF also allows packet authentication and uses IP multicast when sending and receiving packets. The Cisco implementation supports RFC 1253, OSPF management information base (MIB).

The Cisco implementation conforms to the OSPF Version 2 specifications with these key features:

- Definition of stub areas is supported.
- Routes learned through any IP routing protocol can be redistributed into another IP routing protocol. At the intradomain level, this means that OSPF can import routes learned through EIGRP and RIP. OSPF routes can also be exported into RIP.

- Plain text and MD5 authentication among neighboring routers within an area is supported.
- Configurable routing interface parameters include interface output cost, retransmission interval, interface transmit delay, router priority, router dead and hello intervals, and authentication key.
- Virtual links are supported.
- Not-so-stubby-areas (NSSAs) per RFC 1587 are supported.

OSPF typically requires coordination among many internal routers, area border routers (ABRs) connected to multiple areas, and autonomous system boundary routers (ASBRs). The minimum configuration would use all default parameter values, no authentication, and interfaces assigned to areas. If you customize your environment, you must ensure coordinated configuration of all routers.

OSPF for Routed Access



Note OSPF is supported in IP Lite. OSPF for Routed Access supports only one OSPFv2 and one OSPFv3 instance with a combined total of 1000 dynamically learned routes. The IP Lite image provides OSPF for routed access. However, these restrictions are not enforced in this release.

With the typical topology (hub and spoke) in a campus environment, where the wiring closets (spokes) are connected to the distribution switch (hub) that forwards all nonlocal traffic to the distribution layer, the wiring closet switch need not hold a complete routing table. A best practice design, where the distribution switch sends a default route to the wiring closet switch to reach interarea and external routes (OSPF stub or totally stub area configuration) should be used when OSPF for Routed Access is used in the wiring closet.

For more details, see the “High Availability Campus Network Design—Routed Access Layer using EIGRP or OSPF” document.

OSPF Area Parameters

You can optionally configure several OSPF area parameters. These parameters include authentication for password-based protection against unauthorized access to an area, stub areas, and not-so-stubby-areas (NSSAs). Stub areas are areas into which information on external routes is not sent. Instead, the area border router (ABR) generates a default external route into the stub area for destinations outside the autonomous system (AS). An NSSA does not flood all LSAs from the core into the area, but can import AS external routes within the area by redistribution.

Route summarization is the consolidation of advertised addresses into a single summary route to be advertised by other areas. If network numbers are contiguous, you can use the **area range** router configuration command to configure the ABR to advertise a summary route that covers all networks in the range.

Other OSPF Parameters

You can optionally configure other OSPF parameters in router configuration mode.

- Route summarization: When redistributing routes from other protocols. Each route is advertised individually in an external LSA. To help decrease the size of the OSPF link state database, you can use the **summary-address** router configuration command to advertise a single router for all the redistributed routes included in a specified network address and mask.

- **Virtual links:** In OSPF, all areas must be connected to a backbone area. You can establish a virtual link in case of a backbone-continuity break by configuring two Area Border Routers as endpoints of a virtual link. Configuration information includes the identity of the other virtual endpoint (the other ABR) and the nonbackbone link that the two routers have in common (the transit area). Virtual links cannot be configured through a stub area.
- **Default route:** When you specifically configure redistribution of routes into an OSPF routing domain, the route automatically becomes an autonomous system boundary router (ASBR). You can force the ASBR to generate a default route into the OSPF routing domain.
- **Domain Name Server (DNS) names for use in all OSPF `show` privileged EXEC command displays** makes it easier to identify a router than displaying it by router ID or neighbor ID.
- **Default Metrics:** OSPF calculates the OSPF metric for an interface according to the bandwidth of the interface. The metric is calculated as *ref-bw* divided by bandwidth, where *ref* is 10 by default, and bandwidth (*bw*) is specified by the **bandwidth** interface configuration command. For multiple links with high bandwidth, you can specify a larger number to differentiate the cost on those links.
- **Administrative distance** is a rating of the trustworthiness of a routing information source, an integer between 0 and 255, with a higher value meaning a lower trust rating. An administrative distance of 255 means the routing information source cannot be trusted at all and should be ignored. OSPF uses three different administrative distances: routes within an area (interarea), routes to another area (interarea), and routes from another routing domain learned through redistribution (external). You can change any of the distance values.
- **Passive interfaces:** Because interfaces between two devices on an Ethernet represent only one network segment, to prevent OSPF from sending hello packets for the sending interface, you must configure the sending device to be a passive interface. Both devices can identify each other through the hello packet for the receiving interface.
- **Route calculation timers:** You can configure the delay time between when OSPF receives a topology change and when it starts the shortest path first (SPF) calculation and the hold time between two SPF calculations.
- **Log neighbor changes:** You can configure the router to send a syslog message when an OSPF neighbor state changes, providing a high-level view of changes in the router.

LSA Group Pacing

The OSPF LSA group pacing feature allows the router to group OSPF LSAs and pace the refreshing, check-summing, and aging functions for more efficient router use. This feature is enabled by default with a 4-minute default pacing interval, and you will not usually need to modify this parameter. The optimum group pacing interval is inversely proportional to the number of LSAs the router is refreshing, check-summing, and aging. For example, if you have approximately 10,000 LSAs in the database, decreasing the pacing interval would benefit you. If you have a very small database (40 to 100 LSAs), increasing the pacing interval to 10 to 20 minutes might benefit you slightly.

Loopback Interfaces

OSPF uses the highest IP address configured on the interfaces as its router ID. If this interface is down or removed, the OSPF process must recalculate a new router ID and resend all its routing information out its interfaces. If a loopback interface is configured with an IP address, OSPF uses this IP address as its router

ID, even if other interfaces have higher IP addresses. Because loopback interfaces never fail, this provides greater stability. OSPF automatically prefers a loopback interface over other interfaces, and it chooses the highest IP address among all loopback interfaces.

How to Configure OSPF

Default OSPF Configuration

Table 95: Default OSPF Configuration

Feature	Default Setting
Interface parameters	Cost: Retransmit interval: 5 seconds. Transmit delay: 1 second. Priority: 1. Hello interval: 10 seconds. Dead interval: 4 times the hello interval. No authentication. No password specified. MD5 authentication disabled.
Area	Authentication type: 0 (no authentication). Default cost: 1. Range: Disabled. Stub: No stub area defined. NSSA: No NSSA area defined.
Auto cost	100 Mb/s.
Default-information originate	Disabled. When enabled, the default metric setting is 10, and the external route type default is Type 2.
Default metric	Built-in, automatic metric translation, as appropriate for each routing protocol.
Distance OSPF	dist1 (all routes within an area): 110. dist2 (all routes from one area to another): 110. and dist3 (routes from other routing domains): 110.
OSPF database filter	Disabled. All outgoing link-state advertisements (LSAs) are flooded to the interface.
IP OSPF name lookup	Disabled.

Feature	Default Setting
Log adjacency changes	Enabled.
Neighbor	None specified.
Neighbor database filter	Disabled. All outgoing LSAs are flooded to the neighbor.
Network area	Disabled.
Router ID	No OSPF routing process defined.
Summary address	Disabled.
Timers LSA group pacing	240 seconds.
Timers shortest path first (spf)	spf delay: 5 seconds.; spf-holdtime: 10 seconds.
Virtual link	No area ID or router ID defined. Hello interval: 10 seconds. Retransmit interval: 5 seconds. Transmit delay: 1 second. Dead interval: 40 seconds. Authentication key: no key predefined. Message-digest key (MD5): no key predefined.

Configuring Basic OSPF Parameters

To enable OSPF, create an OSPF routing process, specify the range of IP addresses to associate with the routing process, and assign area IDs to be associated with that range.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 2	router ospf process-id Example: Device(config)# router ospf 15	Enables OSPF routing, and enter router configuration mode. The process ID is an internally used identification parameter that is locally assigned and can be any positive integer. Each OSPF routing process has a unique value.

	Command or Action	Purpose
		Note OSPF for Routed Access supports only one OSPFv2 and one OSPFv3 instance with a maximum number of 1000 dynamically learned routes.
Step 3	network <i>address wildcard-mask</i> area <i>area-id</i> Example: Device(config-router)# network 10.1.1.1 255.240.0.0 area 20	Define an interface on which OSPF runs and the area ID for that interface. You can use the wildcard-mask to use a single command to define one or more multiple interfaces to be associated with a specific OSPF area. The area ID can be a decimal value or an IP address.
Step 4	end Example: Device(config-router)# end	Returns to privileged EXEC mode.
Step 5	show ip protocols Example: Device# show ip protocols	Verifies your entries.
Step 6	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Configuring OSPF Interfaces

You can use the **ip ospf** interface configuration commands to modify interface-specific OSPF parameters. You are not required to modify any of these parameters, but some interface parameters (hello interval, dead interval, and authentication key) must be consistent across all routers in an attached network. If you modify these parameters, be sure all routers in the network have compatible values.



Note The **ip ospf** interface configuration commands are all optional.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example:	Enters global configuration mode.

	Command or Action	Purpose
	Device# configure terminal	
Step 2	interface <i>interface-id</i> Example: Device(config)# interface gigabitethernet 1/0/1	Enters interface configuration mode, and specifies the Layer 3 interface to configure.
Step 3	ip ospf cost <i>cost</i> Example: Device(config-if)# ip ospf cost 8	(Optional) Explicitly specifies the cost of sending a packet on the interface.
Step 4	ip ospf retransmit-interval <i>seconds</i> Example: Device(config-if)# ip ospf transmit-interval 10	(Optional) Specifies the number of seconds between link state advertisement transmissions. The range is 1 to 65535 seconds. The default is 5 seconds.
Step 5	ip ospf transmit-delay <i>seconds</i> Example: Device(config-if)# ip ospf transmit-delay 2	(Optional) Sets the estimated number of seconds to wait before sending a link state update packet. The range is 1 to 65535 seconds. The default is 1 second.
Step 6	ip ospf priority <i>number</i> Example: Device(config-if)# ip ospf priority 5	(Optional) Sets priority to help find the OSPF designated router for a network. The range is from 0 to 255. The default is 1.
Step 7	ip ospf hello-interval <i>seconds</i> Example: Device(config-if)# ip ospf hello-interval 12	(Optional) Sets the number of seconds between hello packets sent on an OSPF interface. The value must be the same for all nodes on a network. The range is 1 to 65535 seconds. The default is 10 seconds.
Step 8	ip ospf dead-interval <i>seconds</i> Example: Device(config-if)# ip ospf dead-interval 8	(Optional) Sets the number of seconds after the last device hello packet was seen before its neighbors declare the OSPF router to be down. The value must be the same for all nodes on a network. The range is 1 to 65535 seconds. The default is 4 times the hello interval.
Step 9	ip ospf authentication-key <i>key</i> Example: Device(config-if)# ip ospf authentication-key password	(Optional) Assign a password to be used by neighboring OSPF routers. The password can be any string of keyboard-entered characters up to 8 bytes in length. All neighboring routers on the same network must have the same password to exchange OSPF information.
Step 10	ip ospf message-digest-key <i>keyid md5 key</i> Example:	(Optional) Enables MDS authentication. <ul style="list-style-type: none">• <i>keyid</i>—An identifier from 1 to 255.

	Command or Action	Purpose
	Device(config-if)# ip ospf message digest-key 16 md5 your1pass	<ul style="list-style-type: none"> • <i>key</i>—An alphanumeric password of up to 16 bytes.
Step 11	ip ospf database-filter all out Example: Device(config-if)# ip ospf database-filter all out	(Optional) Block flooding of OSPF LSA packets to the interface. By default, OSPF floods new LSAs over all interfaces in the same area, except the interface on which the LSA arrives.
Step 12	end Example: Device(config)# end	Returns to privileged EXEC mode.
Step 13	show ip ospf interface [interface-name] Example: Device# show ip ospf interface	Displays OSPF-related interface information.
Step 14	show ip ospf neighbor detail Example: Device# show ip ospf neighbor detail	Displays NSF awareness status of neighbor switch. The output matches one of these examples: <ul style="list-style-type: none"> • <i>Options is 0x52</i> <i>LLS Options is 0x1 (LR)</i> When both of these lines appear, the neighbor switch is NSF aware. • <i>Options is 0x42</i>—This means the neighbor switch is not NSF aware.
Step 15	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Configuring OSPF Area Parameters

Before you begin



Note The OSPF **area** router configuration commands are all optional.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 2	router ospf process-id Example: Device(config)# router ospf 109	Enables OSPF routing, and enter router configuration mode.
Step 3	area area-id authentication Example: Device(config-router)# area 1 authentication	(Optional) Allow password-based protection against unauthorized access to the identified area. The identifier can be either a decimal value or an IP address.
Step 4	area area-id authentication message-digest Example: Device(config-router)# area 1 authentication message-digest	(Optional) Enables MD5 authentication on the area.
Step 5	area area-id stub [no-summary] Example: Device(config-router)# area 1 stub	(Optional) Define an area as a stub area. The no-summary keyword prevents an ABR from sending summary link advertisements into the stub area.
Step 6	area area-id nssa [no-redistribution] [default-information-originate] [no-summary] Example: Device(config-router)# area 1 nssa default-information-originate	(Optional) Defines an area as a not-so-stubby-area. Every router within the same area must agree that the area is NSSA. Select one of these keywords: <ul style="list-style-type: none"> • no-redistribution—Select when the router is an NSSA ABR and you want the redistribute command to import routes into normal areas, but not into the NSSA. • default-information-originate—Select on an ABR to allow importing type 7 LSAs into the NSSA. • no-redistribution—Select to not send summary LSAs into the NSSA.
Step 7	area area-id range address mask Example: Device(config-router)# area 1 range 255.240.0.0	(Optional) Specifies an address range for which a single route is advertised. Use this command only with area border routers.

	Command or Action	Purpose
Step 8	end Example: Device(config)# end	Returns to privileged EXEC mode.
Step 9	show ip ospf [<i>process-id</i>] Example: Device# show ip ospf	Displays information about the OSPF routing process in general or for a specific process ID to verify configuration.
Step 10	show ip ospf [<i>process-id</i> [<i>area-id</i>]] database Example: Device# show ip ospf database	Displays lists of information related to the OSPF database for a specific router.
Step 11	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Configuring Other OSPF Parameters

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 2	router ospf <i>process-id</i> Example: Device(config)# router ospf 10	Enables OSPF routing, and enter router configuration mode.
Step 3	summary-address <i>address mask</i> Example: Device(config)# summary-address 10.1.1.1 255.255.255.0	(Optional) Specifies an address and IP subnet mask for redistributed routes so that only one summary route is advertised.
Step 4	area <i>area-id</i> virtual-link <i>router-id</i> [hello-interval <i>seconds</i>] [retransmit-interval <i>seconds</i>] [trans]	(Optional) Establishes a virtual link and set its parameters.

	Command or Action	Purpose
	<p>[[authentication-key <i>key</i>] message-digest-key <i>keyid</i> md5 <i>key</i>]]</p> <p>Example:</p> <pre>Device(config)# area 2 virtual-link 192.168.255.1 hello-interval 5</pre>	
Step 5	<p>default-information originate [always] [metric <i>metric-value</i>] [metric-type <i>type-value</i>] [route-map <i>map-name</i>]</p> <p>Example:</p> <pre>Device(config)# default-information originate metric 100 metric-type 1</pre>	(Optional) Forces the ASBR to generate a default route into the OSPF routing domain. Parameters are all optional.
Step 6	<p>ip ospf name-lookup</p> <p>Example:</p> <pre>Device(config)# ip ospf name-lookup</pre>	(Optional) Configures DNS name lookup. The default is disabled.
Step 7	<p>ip auto-cost reference-bandwidth <i>ref-bw</i></p> <p>Example:</p> <pre>Device(config)# ip auto-cost reference-bandwidth 5</pre>	(Optional) Specifies an address range for which a single route will be advertised. Use this command only with area border routers.
Step 8	<p>distance ospf {[inter-area <i>dist1</i>] [inter-area <i>dist2</i>] [external <i>dist3</i>]}</p> <p>Example:</p> <pre>Device(config)# distance ospf inter-area 150</pre>	(Optional) Changes the OSPF distance values. The default distance for each type of route is 110. The range is 1 to 255.
Step 9	<p>passive-interface <i>type number</i></p> <p>Example:</p> <pre>Device(config)# passive-interface gigabitethernet 1/0/6</pre>	(Optional) Suppresses the sending of hello packets through the specified interface.
Step 10	<p>timers throttle spf <i>spf-delay</i> <i>spf-holdtime</i> <i>spf-wait</i></p> <p>Example:</p> <pre>Device(config)# timers throttle spf 200 100 100</pre>	<p>(Optional) Configures route calculation timers.</p> <ul style="list-style-type: none"> • <i>spf-delay</i>—Delay between receiving a change to SPF calculation. The range is from 1 to 600000 milliseconds. • <i>spf-holdtime</i>—Delay between first and second SPF calculation. The range is form 1 to 600000 in milliseconds.

	Command or Action	Purpose
		<ul style="list-style-type: none"> <i>spf-wait</i>—Maximum wait time in milliseconds for SPF calculations. The range is from 1 to 600000 in milliseconds.
Step 11	ospf log-adj-changes Example: Device(config)# ospf log-adj-changes	(Optional) Sends syslog message when a neighbor state changes.
Step 12	end Example: Device(config)# end	Returns to privileged EXEC mode.
Step 13	show ip ospf [process-id [area-id]] database Example: Device# show ip ospf database	Displays lists of information related to the OSPF database for a specific router.
Step 14	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Changing LSA Group Pacing

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 2	router ospf process-id Example: Device(config)# router ospf 25	Enables OSPF routing, and enter router configuration mode.
Step 3	timers lsa-group-pacing seconds Example: Device(config-router)# timers lsa-group-pacing 15	Changes the group pacing of LSAs.

	Command or Action	Purpose
Step 4	end Example: Device(config)# end	Returns to privileged EXEC mode.
Step 5	show running-config Example: Device# show running-config	Verifies your entries.
Step 6	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Configuring a Loopback Interface

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 2	interface loopback 0 Example: Device(config)# interface loopback 0	Creates a loopback interface, and enter interface configuration mode.
Step 3	ip address address mask Example: Device(config-if)# ip address 10.1.1.5 255.255.240.0	Assign an IP address to this interface.
Step 4	end Example: Device(config)# end	Returns to privileged EXEC mode.

	Command or Action	Purpose
Step 5	show ip interface Example: Device# show ip interface	Verifies your entries.
Step 6	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Monitoring OSPF

You can display specific statistics such as the contents of IP routing tables, caches, and databases.

Table 96: Show IP OSPF Statistics Commands

show ip ospf [<i>process-id</i>]	Displays general information about OSPF routing processes.
show ip ospf [<i>process-id</i>] database [router] [<i>link-state-id</i>] show ip ospf [<i>process-id</i>] database [router] [self-originate] show ip ospf [<i>process-id</i>] database [router] [adv-router [<i>ip-address</i>]] show ip ospf [<i>process-id</i>] database [network] [<i>link-state-id</i>] show ip ospf [<i>process-id</i>] database [summary] [<i>link-state-id</i>] show ip ospf [<i>process-id</i>] database [asbr-summary] [<i>link-state-id</i>] show ip ospf [<i>process-id</i>] database [external] [<i>link-state-id</i>] show ip ospf [<i>process-id area-id</i>] database [database-summary]	Displays lists of information related to the OSPF database.
show ip ospf border-routes	Displays the internal OSPF routing ABR and ASBR table entries.
show ip ospf interface [<i>interface-name</i>]	Displays OSPF-related interface information.
show ip ospf neighbor [<i>interface-name</i>] [<i>neighbor-id</i>] detail	Displays OSPF interface neighbor information.

<code>show ip ospf virtual-links</code>	Displays OSPF-related virtual links information.
---	--

Configuration Examples for OSPF

Example: Configuring Basic OSPF Parameters

This example shows how to configure an OSPF routing process and assign it a process number of 109:

```
Device(config)# router ospf 109
Device(config-router)# network 131.108.0.0 255.255.255.0 area 24
```

Information About EIGRP

Enhanced IGRP (EIGRP) is a Cisco proprietary enhanced version of the IGRP. EIGRP uses the same distance vector algorithm and distance information as IGRP; however, the convergence properties and the operating efficiency of EIGRP are significantly improved.

The convergence technology employs an algorithm referred to as the Diffusing Update Algorithm (DUAL), which guarantees loop-free operation at every instant throughout a route computation and allows all devices involved in a topology change to synchronize at the same time. Routers that are not affected by topology changes are not involved in recomputations.

IP EIGRP provides increased network width. With RIP, the largest possible width of your network is 15 hops. Because the EIGRP metric is large enough to support thousands of hops, the only barrier to expanding the network is the transport-layer hop counter. EIGRP increments the transport control field only when an IP packet has traversed 15 routers and the next hop to the destination was learned through EIGRP. When a RIP route is used as the next hop to the destination, the transport control field is incremented as usual.

EIGRP Stub Routing

The EIGRP stub routing feature reduces resource utilization by moving routed traffic closer to the end user.



Note The device uses EIGRP stub routing at the access layer to eliminate the need for other types of routing advertisements.

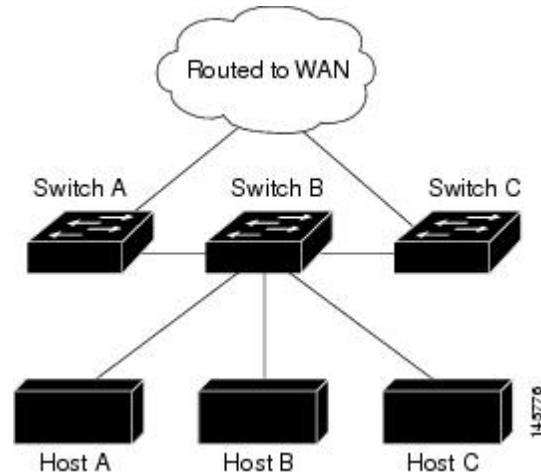
In a network using EIGRP stub routing, the only allowable route for IP traffic to the user is through a device that is configured with EIGRP stub routing. The device sends the routed traffic to interfaces that are configured as user interfaces or are connected to other devices.

When using EIGRP stub routing, you need to configure the distribution and remote routers to use EIGRP and to configure only the device as a stub. Only specified routes are propagated from the device. The device responds to all queries for summaries, connected routes, and routing updates.

Any neighbor that receives a packet informing it of the stub status does not query the stub router for any routes, and a router that has a stub peer does not query that peer. The stub router depends on the distribution router to send the proper updates to all peers.

In the figure given below, device B is configured as an EIGRP stub router. Devices A and C are connected to the rest of the WAN. Device B advertises connected, static, redistribution, and summary routes to Device A and C. Device B does not advertise any routes learned from Device A (and the reverse).

Figure 84: EIGRP Stub Router Configuration



For more information about EIGRP stub routing, see “Configuring EIGRP Stub Routing” section of the *Cisco IOS IP Configuration Guide, Volume 2 of 3: Routing Protocols*.

Configuring Unicast Reverse Path Forwarding

The unicast reverse path forwarding (unicast RPF) feature helps to mitigate problems that are caused by the introduction of malformed or forged (spoofed) IP source addresses into a network by discarding IP packets that lack a verifiable IP source address. For example, a number of common types of denial-of-service (DoS) attacks, including Smurf and Tribal Flood Network (TFN), can take advantage of forged or rapidly changing source IP addresses to allow attackers to thwart efforts to locate or filter the attacks. For Internet service providers (ISPs) that provide public access, Unicast RPF deflects such attacks by forwarding only packets that have source addresses that are valid and consistent with the IP routing table. This action protects the network of the ISP, its customer, and the rest of the Internet.



Note • Unicast RPF is supported in IP Lite.

For detailed IP unicast RPF configuration information, see the *Other Security Features* chapter in the *Cisco IOS Security Configuration Guide*.

Protocol-Independent Features

This section describes IP routing protocol-independent features that are available on switches running the IP Lite feature set. For a complete description of the IP routing protocol-independent commands in this chapter, see the “IP Routing Protocol-Independent Commands” chapter of the *Cisco IOS IP Command Reference, Volume 2 of 3: Routing Protocols*.

Distributed Cisco Express Forwarding

Information About Cisco Express Forwarding

Cisco Express Forwarding (CEF) is a Layer 3 IP switching technology used to optimize network performance. CEF implements an advanced IP look-up and forwarding algorithm to deliver maximum Layer 3 switching performance. CEF is less CPU-intensive than fast switching route caching, allowing more CPU processing power to be dedicated to packet forwarding. In a switch stack, the hardware uses distributed CEF (dCEF) in the stack. In dynamic networks, fast switching cache entries are frequently invalidated because of routing changes, which can cause traffic to be process switched using the routing table, instead of fast switched using the route cache. CEF and dCEF use the Forwarding Information Base (FIB) lookup table to perform destination-based switching of IP packets.

The two main components in CEF and dCEF are the distributed FIB and the distributed adjacency tables.

- The FIB is similar to a routing table or information base and maintains a mirror image of the forwarding information in the IP routing table. When routing or topology changes occur in the network, the IP routing table is updated, and those changes are reflected in the FIB. The FIB maintains next-hop address information based on the information in the IP routing table. Because the FIB contains all known routes that exist in the routing table, CEF eliminates route cache maintenance, is more efficient for switching traffic, and is not affected by traffic patterns.
- Nodes in the network are said to be adjacent if they can reach each other with a single hop across a link layer. CEF uses adjacency tables to prepend Layer 2 addressing information. The adjacency table maintains Layer 2 next-hop addresses for all FIB entries.

Because the switch or switch stack uses Application Specific Integrated Circuits (ASICs) to achieve Gigabit-speed line rate IP traffic, CEF or dCEF forwarding applies only to the software-forwarding path, that is, traffic that is forwarded by the CPU.

How to Configure Cisco Express Forwarding

CEF or distributed CEF is enabled globally by default. If for some reason it is disabled, you can re-enable it by using the **ip cef** or **ip cef distributed** global configuration command.

The default configuration is CEF or dCEF enabled on all Layer 3 interfaces. Entering the **no ip route-cache cef** interface configuration command disables CEF for traffic that is being forwarded by software. This command does not affect the hardware forwarding path. Disabling CEF and using the **debug ip packet detail** privileged EXEC command can be useful to debug software-forwarded traffic. To enable CEF on an interface for the software-forwarding path, use the **ip route-cache cef** interface configuration command.



Caution Although the **no ip route-cache cef** interface configuration command to disable CEF on an interface is visible in the CLI, we strongly recommend that you do not disable CEF or dCEF on interfaces except for debugging purposes.

To enable CEF or dCEF globally and on an interface for software-forwarded traffic if it has been disabled:

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 2	ip cef Example: Device(config)# ip cef	Enables CEF operation on a non-stacking switch. Go to Step 4.
Step 3	ip cef distributed Example: Device(config)# ip cef distributed	Enables CEF operation on a active switch.
Step 4	interface <i>interface-id</i> Example: Device(config)# interface gigabitethernet 1/0/1	Enters interface configuration mode, and specifies the Layer 3 interface to configure.
Step 5	ip route-cache cef Example: Device(config-if)# ip route-cache cef	Enables CEF on the interface for software-forwarded traffic.
Step 6	end Example: Device(config-if)# end	Returns to privileged EXEC mode.
Step 7	show ip cef Example: Device# show ip cef	Displays the CEF status on all interfaces.
Step 8	show cef linecard [detail] Example: Device# show cef linecard detail	(Optional) Displays CEF-related interface information on a non-stacking switch.
Step 9	show cef linecard [<i>slot-number</i>] [detail] Example: Device# show cef linecard 5 detail	(Optional) Displays CEF-related interface information on a switch by stack member for all switches in the stack or for the specified switch. (Optional) For <i>slot-number</i> , enter the stack member switch number.

	Command or Action	Purpose
Step 10	show cef interface [<i>interface-id</i>] Example: Device# show cef interface gigabitethernet 1/0/1	Displays detailed CEF information for all interfaces or the specified interface.
Step 11	show adjacency Example: Device# show adjacency	Displays CEF adjacency table information.
Step 12	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Number of Equal-Cost Routing Paths

Information About Equal-Cost Routing Paths

When a router has two or more routes to the same network with the same metrics, these routes can be thought of as having an equal cost. The term parallel path is another way to see occurrences of equal-cost routes in a routing table. If a router has two or more equal-cost paths to a network, it can use them concurrently. Parallel paths provide redundancy in case of a circuit failure and also enable a router to load balance packets over the available paths for more efficient use of available bandwidth. Equal-cost routes are supported across switches in a stack.

Even though the router automatically learns about and configures equal-cost routes, you can control the maximum number of parallel paths supported by an IP routing protocol in its routing table. Although the switch software allows a maximum of 32 equal-cost routes, the switch hardware will never use more than 16 paths per route.

How to Configure Equal-Cost Routing Paths

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 2	router {rip ospf eigrp} Example: Device(config)# router eigrp 10	Enters router configuration mode.

	Command or Action	Purpose
Step 3	maximum-paths <i>maximum</i> Example: Device(config-router)# maximum-paths 2	Sets the maximum number of parallel paths for the protocol routing table. The range is from 1 to 16; the default is 4 for most IP routing protocols.
Step 4	end Example: Device(config-router)# end	Returns to privileged EXEC mode.
Step 5	show ip protocols Example: Device# show ip protocols	Verifies the setting in the <i>Maximum path</i> field.
Step 6	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Static Unicast Routes

Information About Static Unicast Routes

Static unicast routes are user-defined routes that cause packets moving between a source and a destination to take a specified path. Static routes can be important if the router cannot build a route to a particular destination and are useful for specifying a gateway of last resort to which all unroutable packets are sent.

The switch retains static routes until you remove them. However, you can override static routes with dynamic routing information by assigning administrative distance values. Each dynamic routing protocol has a default administrative distance, as listed in Table 41-16. If you want a static route to be overridden by information from a dynamic routing protocol, set the administrative distance of the static route higher than that of the dynamic protocol.

Table 97: Dynamic Routing Protocol Default Administrative Distances

Route Source	Default Distance
Connected interface	0
Static route	1
Enhanced IRGP summary route	5
Internal Enhanced IGRP	90
IGRP	100
OSPF	110

Route Source	Default Distance
Unknown	225

Static routes that point to an interface are advertised through RIP, IGRP, and other dynamic routing protocols, whether or not static **redistribute** router configuration commands were specified for those routing protocols. These static routes are advertised because static routes that point to an interface are considered in the routing table to be connected and hence lose their static nature. However, if you define a static route to an interface that is not one of the networks defined in a network command, no dynamic routing protocols advertise the route unless a **redistribute** static command is specified for these protocols.

When an interface goes down, all static routes through that interface are removed from the IP routing table. When the software can no longer find a valid next hop for the address specified as the forwarding router's address in a static route, the static route is also removed from the IP routing table.

Configuring Static Unicast Routes

Static unicast routes are user-defined routes that cause packets moving between a source and a destination to take a specified path. Static routes can be important if the router cannot build a route to a particular destination and are useful for specifying a gateway of last resort to which all unroutable packets are sent.

Follow these steps to configure a static route:

Procedure

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	ip route prefix mask {address interface} [distance] Example: Device(config)# ip route prefix mask gigabitethernet 1/0/4	Establish a static route.
Step 4	end Example: Device(config)# end	Returns to privileged EXEC mode.

	Command or Action	Purpose
Step 5	show ip route Example: Device# <code>show ip route</code>	Displays the current state of the routing table to verify the configuration.
Step 6	copy running-config startup-config Example: Device# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

What to do next

Use the **no ip route** *prefix mask {address| interface}* global configuration command to remove a static route. The device retains static routes until you remove them.

Default Routes and Networks

Information About Default Routes and Networks

A router might not be able to learn the routes to all other networks. To provide complete routing capability, you can use some routers as smart routers and give the remaining routers default routes to the smart router. (Smart routers have routing table information for the entire internetwork.) These default routes can be dynamically learned or can be configured in the individual routers. Most dynamic interior routing protocols include a mechanism for causing a smart router to generate dynamic default information that is then forwarded to other routers.

If a router has a directly connected interface to the specified default network, the dynamic routing protocols running on that device generate a default route. In RIP, it advertises the pseudonetwork 0.0.0.0.

A router that is generating the default for a network also might need a default of its own. One way a router can generate its own default is to specify a static route to the network 0.0.0.0 through the appropriate device.

When default information is passed through a dynamic routing protocol, no further configuration is required. The system periodically scans its routing table to choose the optimal default network as its default route. In IGRP networks, there might be several candidate networks for the system default. Cisco routers use administrative distance and metric information to set the default route or the gateway of last resort.

If dynamic default information is not being passed to the system, candidates for the default route are specified with the **ip default-network** global configuration command. If this network appears in the routing table from any source, it is flagged as a possible choice for the default route. If the router has no interface on the default network, but does have a path to it, the network is considered as a possible candidate, and the gateway to the best default path becomes the gateway of last resort.

How to Configure Default Routes and Networks

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode.
Step 2	ip default-network <i>network number</i> Example: Device(config)# <code>ip default-network 1</code>	Specifies a default network.
Step 3	end Example: Device(config)# <code>end</code>	Returns to privileged EXEC mode.
Step 4	show ip route Example: Device# <code>show ip route</code>	Displays the selected default route in the gateway of last resort display.
Step 5	copy running-config startup-config Example: Device# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Route Maps to Redistribute Routing Information

Information About Route Maps

The switch can run multiple routing protocols simultaneously, and it can redistribute information from one routing protocol to another. Redistributing information from one routing protocol to another applies to all supported IP-based routing protocols.

You can also conditionally control the redistribution of routes between routing domains by defining enhanced packet filters or route maps between the two domains. The **match** and **set** route-map configuration commands define the condition portion of a route map. The **match** command specifies that a criterion must be matched. The **set** command specifies an action to be taken if the routing update meets the conditions defined by the match command. Although redistribution is a protocol-independent feature, some of the **match** and **set** route-map configuration commands are specific to a particular protocol.

One or more **match** commands and one or more **set** commands follow a **route-map** command. If there are no **match** commands, everything matches. If there are no **set** commands, nothing is done, other than the match. Therefore, you need at least one **match** or **set** command.



Note A route map with no **set** route-map configuration commands is sent to the CPU, which causes high CPU utilization.

You can also identify route-map statements as **permit** or **deny**. If the statement is marked as a deny, the packets meeting the match criteria are sent back through the normal forwarding channels (destination-based routing). If the statement is marked as permit, set clauses are applied to packets meeting the match criteria. Packets that do not meet the match criteria are forwarded through the normal routing channel.

How to Configure a Route Map

Although each of Steps 3 through 14 in the following section is optional, you must enter at least one **match** route-map configuration command and one **set** route-map configuration command.



Note The keywords are the same as defined in the procedure to control the route distribution.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 2	route-map map-tag [permit deny] [sequence number] Example: Device(config)# route-map rip-to-ospf permit 4	Defines any route maps used to control redistribution and enter route-map configuration mode. <i>map-tag</i> —A meaningful name for the route map. The redistribute router configuration command uses this name to reference this route map. Multiple route maps might share the same map tag name. (Optional) If permit is specified and the match criteria are met for this route map, the route is redistributed as controlled by the set actions. If deny is specified, the route is not redistributed. <i>sequence number</i> (Optional)— Number that indicates the position a new route map is to have in the list of route maps already configured with the same name.
Step 3	match metric metric-value Example: Device(config-route-map)# match metric 2000	Matches the specified route metric. The <i>metric-value</i> can be an EIGRP metric with a specified value from 0 to 4294967295.

	Command or Action	Purpose
Step 4	<p>match ip next-hop {<i>access-list-number</i> <i>access-list-name</i>} [...<i>access-list-number</i> ...<i>access-list-name</i>]</p> <p>Example:</p> <pre>Device(config-route-map)# match ip next-hop 8 45</pre>	Matches a next-hop router address passed by one of the access lists specified (numbered from 1 to 199).
Step 5	<p>match tag tag value [...tag-value]</p> <p>Example:</p> <pre>Device(config-route-map)# match tag 3500</pre>	Matches the specified tag value in a list of one or more route tag values. Each can be an integer from 0 to 4294967295.
Step 6	<p>match interface type number [...type number]</p> <p>Example:</p> <pre>Device(config-route-map)# match interface gigabitethernet 1/0/1</pre>	Matches the specified next hop route out one of the specified interfaces.
Step 7	<p>match ip route-source {<i>access-list-number</i> <i>access-list-name</i>} [...<i>access-list-number</i> ...<i>access-list-name</i>]</p> <p>Example:</p> <pre>Device(config-route-map)# match ip route-source 10 30</pre>	Matches the address specified by the specified advertised access lists.
Step 8	<p>match route-type {<i>local</i> <i>internal</i> <i>external</i> [<i>type-1</i> <i>type-2</i>]}</p> <p>Example:</p> <pre>Device(config-route-map)# match route-type local</pre>	Matches the specified route-type : <ul style="list-style-type: none"> • internal—OSPF intra-area and interarea routes or EIGRP internal routes. • external—OSPF external routes (Type 1 or Type 2) or EIGRP external routes.
Step 9	<p>set metric metric value</p> <p>Example:</p> <pre>Device(config-route-map)# set metric 100</pre>	Sets the metric value to give the redistributed routes (for EIGRP only). The <i>metric value</i> is an integer from -294967295 to 294967295.
Step 10	<p>set metric bandwidth delay reliability loading mtu</p> <p>Example:</p> <pre>Device(config-route-map)# set metric 10000 10 255 1 1500</pre>	Sets the metric value to give the redistributed routes (for EIGRP only): <ul style="list-style-type: none"> • <i>bandwidth</i>—Metric value or IGRP bandwidth of the route in kilobits per second in the range 0 to 4294967295 • <i>delay</i>—Route delay in tens of microseconds in the range 0 to 4294967295. • <i>reliability</i>—Likelihood of successful packet transmission expressed as a number between 0 and

	Command or Action	Purpose
		<p>255, where 255 means 100 percent reliability and 0 means no reliability.</p> <ul style="list-style-type: none"> • <i>loading</i>—Effective bandwidth of the route expressed as a number from 0 to 255 (255 is 100 percent loading). • <i>mtu</i>—Minimum maximum transmission unit (MTU) size of the route in bytes in the range 0 to 4294967295.
Step 11	<p>set metric-type {type-1 type-2}</p> <p>Example:</p> <pre>Device(config-route-map)# set metric-type type-2</pre>	Sets the OSPF external metric type for redistributed routes.
Step 12	<p>end</p> <p>Example:</p> <pre>Device(config-route-map)# end</pre>	Returns to privileged EXEC mode.
Step 13	<p>show route-map</p> <p>Example:</p> <pre>Device# show route-map</pre>	Displays all route maps configured or only the one specified to verify configuration.
Step 14	<p>copy running-config startup-config</p> <p>Example:</p> <pre>Device# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

How to Control Route Distribution

Although each of Steps 3 through 14 in the following section is optional, you must enter at least one **match** route-map configuration command and one **set** route-map configuration command.



Note The keywords are the same as defined in the procedure to configure the route map for redistribution.

The metrics of one routing protocol do not necessarily translate into the metrics of another. For example, the RIP metric is a hop count, and the IGRP metric is a combination of five qualities. In these situations, an artificial metric is assigned to the redistributed route. Uncontrolled exchanging of routing information between different routing protocols can create routing loops and seriously degrade network operation.

If you have not defined a default redistribution metric that replaces metric conversion, some automatic metric translations occur between routing protocols:

- RIP can automatically redistribute static routes. It assigns static routes a metric of 1 (directly connected).

- Any protocol can redistribute other routing protocols if a default mode is in effect.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 2	router { rip ospf eigrp } Example: Device(config)# router eigrp 10	Enters router configuration mode.
Step 3	redistribute protocol [process-id] {level-1 level-1-2 level-2} [metric metric-value] [metric-type type-value] [match internal external type-value] [tag tag-value] [route-map map-tag] [weight weight] [subnets] Example: Device(config-router)# redistribute eigrp 1	Redistributes routes from one routing protocol to another routing protocol. If no route-maps are specified, all routes are redistributed. If the keyword route-map is specified with no <i>map-tag</i> , no routes are distributed.
Step 4	default-metric number Example: Device(config-router)# default-metric 1024	Cause the current routing protocol to use the same metric value for all redistributed routes (RIP and OSPF).
Step 5	default-metric bandwidth delay reliability loading mtu Example: Device(config-router)# default-metric 1000 100 250 100 1500	Cause the EIGRP routing protocol to use the same metric value for all non-EIGRP redistributed routes.
Step 6	end Example: Device(config-router)# end	Returns to privileged EXEC mode.
Step 7	show route-map Example: Device# show route-map	Displays all route maps configured or only the one specified to verify configuration.
Step 8	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Policy-Based Routing

Information About Policy-Based Routing

You can use policy-based routing (PBR) to configure a defined policy for traffic flows. By using PBR, you can have more control over routing by reducing the reliance on routes derived from routing protocols. PBR can specify and implement routing policies that allow or deny paths based on:

- Identity of a particular end system
- Application
- Protocol

You can use PBR to provide equal-access and source-sensitive routing, routing based on interactive versus batch traffic, or routing based on dedicated links. For example, you could transfer stock records to a corporate office on a high-bandwidth, high-cost link for a short time while transmitting routine application data such as e-mail over a low-bandwidth, low-cost link.

With PBR, you classify traffic using access control lists (ACLs) and then make traffic go through a different path. PBR is applied to incoming packets. All packets received on an interface with PBR enabled are passed through route maps. Based on the criteria defined in the route maps, packets are forwarded (routed) to the appropriate next hop.

- Route map statement marked as permit is processed as follows:
 - A match command can match on length or multiple ACLs. A route map statement can contain multiple match commands. Logical or algorithm function is performed across all the match commands to reach a permit or deny decision.

For example:

```
match length A B
match ip address acl1 acl2
match ip address acl3
```

A packet is permitted if it is permitted by match length A B or acl1 or acl2 or acl3

- If the decision reached is permit, then the action specified by the set command is applied on the packet .
- If the decision reached is deny, then the PBR action (specified in the set command) is not applied. Instead the processing logic moves forward to look at the next route-map statement in the sequence (the statement with the next higher sequence number). If no next statement exists, PBR processing terminates, and the packet is routed using the default IP routing table.
- For PBR, route-map statements marked as deny are not supported.

You can use standard IP ACLs to specify match criteria for a source address or extended IP ACLs to specify match criteria based on an application, a protocol type, or an end station. The process proceeds through the route map until a match is found. If no match is found, normal destination-based routing occurs. There is an implicit deny at the end of the list of match statements.

If match clauses are satisfied, you can use a set clause to specify the IP addresses identifying the next hop router in the path.

How to Configure PBR

- To use PBR, you must have the IP Lite feature set enabled on the switch or active stack.
- Multicast traffic is not policy-routed. PBR applies only to unicast traffic.
- You can enable PBR on a routed port or an SVI.
- The switch supports PBR based on match length.
- You can apply a policy route map to an EtherChannel port channel in Layer 3 mode, but you cannot apply a policy route map to a physical interface that is a member of the EtherChannel. If you try to do so, the command is rejected. When a policy route map is applied to a physical interface, that interface cannot become a member of an EtherChannel.
- You can define a maximum of 128 IP policy route maps on the switch or switch stack.
- You can define a maximum of 512 access control entries (ACEs) for PBR on the switch or switch stack.
- When configuring match criteria in a route map, follow these guidelines:
 - Do not match ACLs that permit packets destined for a local address. PBR would forward these packets, which could cause ping or Telnet failure or route protocol flapping.
- VRF and PBR are mutually exclusive on a switch interface. You cannot enable VRF when PBR is enabled on an interface. The reverse is also true, you cannot enable PBR when VRF is enabled on an interface.
- The number of hardware entries used by PBR depends on the route map itself, the ACLs used, and the order of the ACLs and route-map entries.
- PBR based on TOS, DSCP and IP Precedence are not supported.
- Set interface, set default next-hop and set default interface are not supported.
- **ip next-hop recursive** and **ip next-hop verify availability** features are not available and the next-hop should be directly connected.
- Policy-maps with no set actions are supported. Matching packets are routed normally.
- Policy-maps with no match clauses are supported. Set actions are applied to all packets.

By default, PBR is disabled on the switch. To enable PBR, you must create a route map that specifies the match criteria and the resulting action. Then, you must enable PBR for that route map on an interface. All packets arriving on the specified interface matching the match clauses are subject to PBR.

Packets that are generated by the switch, or local packets, are not normally policy-routed. When you globally enable local PBR on the switch, all packets that originate on the switch are subject to local PBR. Local PBR is disabled by default.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters global configuration mode.

	Command or Action	Purpose
Step 2	<p>route-map <i>map-tag</i> [permit] [<i>sequence number</i>]</p> <p>Example:</p> <pre>Device(config)# route-map pbr-map permit</pre>	<p>Defines route maps that are used to control where packets are output, and enters route-map configuration mode.</p> <ul style="list-style-type: none"> • <i>map-tag</i> – A meaningful name for the route map. The ip policy route-map interface configuration command uses this name to reference the route map. Multiple route-map statements with the same map tag define a single route map. • (Optional) permit – If permit is specified and the match criteria are met for this route map, the route is policy routed as defined by the set actions. • (Optional) <i>sequence number</i> – The sequence number shows the position of the route-map statement in the given route map.
Step 3	<p>match ip address {<i>access-list-number</i> <i>access-list-name</i>} [<i>access-list-number</i> ...<i>access-list-name</i>]</p> <p>Example:</p> <pre>Device(config-route-map)# match ip address 110 140</pre>	<p>Matches the source and destination IP addresses that are permitted by one or more standard or extended access lists. ACLs can match on more than one source and destination IP address.</p> <p>If you do not specify a match command, the route map is applicable to all packets.</p>
Step 4	<p>match length <i>min max</i></p> <p>Example:</p> <pre>Device(config-route-map)# match length 64 1500</pre>	Matches the length of the packet.
Step 5	<p>set ip next-hop <i>ip-address</i> [...<i>ip-address</i>]</p> <p>Example:</p> <pre>Device(config-route-map)# set ip next-hop 10.1.6.2</pre>	Specifies the action to be taken on the packets that match the criteria. Sets next hop to which to route the packet (the next hop must be adjacent).
Step 6	<p>exit</p> <p>Example:</p> <pre>Device(config-route-map)# exit</pre>	Returns to global configuration mode.
Step 7	<p>interface <i>interface-id</i></p> <p>Example:</p> <pre>Device(config)# interface gigabitethernet 1/0/1</pre>	Enters interface configuration mode, and specifies the interface to be configured.
Step 8	<p>ip policy route-map <i>map-tag</i></p> <p>Example:</p> <pre>Device(config-if)# ip policy route-map pbr-map</pre>	Enables PBR on a Layer 3 interface, and identify the route map to use. You can configure only one route map on an interface. However, you can have multiple route map entries with different sequence numbers. These entries are evaluated in the order of sequence number until the first match. If there is no match, packets are routed as usual.

	Command or Action	Purpose
Step 9	ip route-cache policy Example: Device(config-if)# ip route-cache policy	(Optional) Enables fast-switching PBR. You must enable PBR before enabling fast-switching PBR.
Step 10	exit Example: Device(config-if)# exit	Returns to global configuration mode.
Step 11	ip local policy route-map map-tag Example: Device(config)# ip local policy route-map local-pbr	(Optional) Enables local PBR to perform policy-based routing on packets originating at the switch. This applies to packets generated by the switch, and not to incoming packets.
Step 12	end Example: Device(config)# end	Returns to privileged EXEC mode.
Step 13	show route-map [map-name] Example: Device# show route-map	(Optional) Displays all the route maps configured or only the one specified to verify configuration.
Step 14	show ip policy Example: Device# show ip policy	(Optional) Displays policy route maps attached to the interface.
Step 15	show ip local policy Example: Device# show ip local policy	(Optional) Displays whether or not local policy routing is enabled and, if so, the route map being used.

Filtering Routing Information

You can filter routing protocol information by performing the tasks described in this section.



Note When routes are redistributed between OSPF processes, no OSPF metrics are preserved.

Setting Passive Interfaces

To prevent other routers on a local network from dynamically learning about routes, you can use the **passive-interface** router configuration command to keep routing update messages from being sent through a router interface. When you use this command in the OSPF protocol, the interface address you specify as passive appears as a stub network in the OSPF domain. OSPF routing information is neither sent nor received through the specified router interface.

In networks with many interfaces, to avoid having to manually set them as passive, you can set all interfaces to be passive by default by using the **passive-interface default** router configuration command and manually setting interfaces where adjacencies are desired.

Use a network monitoring privileged EXEC command such as **show ip ospf interface** to verify the interfaces that you enabled as passive, or use the **show ip interface** privileged EXEC command to verify the interfaces that you enabled as active.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 2	router { rip ospf eigrp } Example: Device(config)# router ospf	Enters router configuration mode.
Step 3	passive-interface <i>interface-id</i> Example: Device(config-router)# passive-interface gigabitethernet 1/0/1	Suppresses sending routing updates through the specified Layer 3 interface.
Step 4	passive-interface default Example: Device(config-router)# passive-interface default	(Optional) Sets all interfaces as passive by default.
Step 5	no passive-interface <i>interface type</i> Example: Device(config-router)# no passive-interface gigabitethernet1/0/3 gigabitethernet 1/0/5	(Optional) Activates only those interfaces that need to have adjacencies sent.
Step 6	network <i>network-address</i> Example: Device(config-router)# network 10.1.1.1	(Optional) Specifies the list of networks for the routing process. The <i>network-address</i> is an IP address.
Step 7	end Example: Device(config-router)# end	Returns to privileged EXEC mode.

	Command or Action	Purpose
Step 8	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Controlling Advertising and Processing in Routing Updates

You can use the **distribute-list** router configuration command with access control lists to suppress routes from being advertised in routing updates and to prevent other routers from learning one or more routes. When used in OSPF, this feature applies to only external routes, and you cannot specify an interface name.

You can also use a **distribute-list** router configuration command to avoid processing certain routes listed in incoming updates. (This feature does not apply to OSPF.)

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 2	router { rip eigrp } Example: Device(config)# router eigrp 10	Enters router configuration mode.
Step 3	distribute-list {access-list-number access-list-name} out <i>[interface-name routing process autonomous-system-number]</i> Example: Device(config-router)# distribute-list 120 out gigabitethernet 1/0/7	Permits or denies routes from being advertised in routing updates, depending upon the action listed in the access list.
Step 4	distribute-list {access-list-number access-list-name} in <i>[type-number]</i> Example: Device(config-router)# distribute-list 125 in	Suppresses processing in routes listed in updates.
Step 5	end Example: Device(config-router)# end	Returns to privileged EXEC mode.

	Command or Action	Purpose
Step 6	copy running-config startup-config Example: <pre>Device# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Filtering Sources of Routing Information

Because some routing information might be more accurate than others, you can use filtering to prioritize information coming from different sources. An administrative distance is a rating of the trustworthiness of a routing information source, such as a router or group of routers. In a large network, some routing protocols can be more reliable than others. By specifying administrative distance values, you enable the router to intelligently discriminate between sources of routing information. The router always picks the route whose routing protocol has the lowest administrative distance.

Because each network has its own requirements, there are no general guidelines for assigning administrative distances.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>Device# configure terminal</pre>	Enters global configuration mode.
Step 2	router { rip ospf eigrp } Example: <pre>Device(config)# router eigrp 10</pre>	Enters router configuration mode.
Step 3	distance weight {ip-address {ip-address mask}} [ip access list] Example: <pre>Device(config-router)# distance 50 10.1.5.1</pre>	Defines an administrative distance. <i>weight</i> —The administrative distance as an integer from 10 to 255. Used alone, <i>weight</i> specifies a default administrative distance that is used when no other specification exists for a routing information source. Routes with a distance of 255 are not installed in the routing table. (Optional) <i>ip access list</i> —An IP standard or extended access list to be applied to incoming routing updates.
Step 4	end Example: <pre>Device(config-router)# end</pre>	Returns to privileged EXEC mode.
Step 5	show ip protocols Example:	Displays the default administrative distance for a specified routing process.

	Command or Action	Purpose
	Device# show ip protocols	
Step 6	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Managing Authentication Keys

Key management is a method of controlling authentication keys used by routing protocols. Not all protocols can use key management. Authentication keys are available for EIGRP and RIP Version 2.

Prerequisites

Before you manage authentication keys, you must enable authentication. See the appropriate protocol section to see how to enable authentication for that protocol. To manage authentication keys, define a key chain, identify the keys that belong to the key chain, and specify how long each key is valid. Each key has its own key identifier (specified with the **key number** key chain configuration command), which is stored locally. The combination of the key identifier and the interface associated with the message uniquely identifies the authentication algorithm and Message Digest 5 (MD5) authentication key in use.

How to Configure Authentication Keys

You can configure multiple keys with life times. Only one authentication packet is sent, regardless of how many valid keys exist. The software examines the key numbers in order from lowest to highest, and uses the first valid key it encounters. The lifetimes allow for overlap during key changes. Note that the router must know these lifetimes.

Procedure

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 2	key chain <i>name-of-chain</i> Example: Device(config)# key chain key10	Identifies a key chain, and enter key chain configuration mode.
Step 3	key number Example: Device(config-keychain)# key 2000	Identifies the key number. The range is 0 to 2147483647.

	Command or Action	Purpose
Step 4	<p>key-string <i>text</i></p> <p>Example:</p> <pre>Device(config-keychain)# key-string Room 20, 10th floor</pre>	Identifies the key string. The string can contain from 1 to 80 uppercase and lowercase alphanumeric characters, but the first character cannot be a number.
Step 5	<p>accept-lifetime <i>start-time</i> {infinite <i>end-time</i> duration <i>seconds</i>}</p> <p>Example:</p> <pre>Device(config-keychain)# accept-lifetime 12:30:00 Jan 25 1009 infinite</pre>	<p>(Optional) Specifies the time period during which the key can be received.</p> <p>The <i>start-time</i> and <i>end-time</i> syntax can be either <i>hh:mm:ss Month date year</i> or <i>hh:mm:ss date Month year</i>. The default is forever with the default <i>start-time</i> and the earliest acceptable date as January 1, 1993. The default <i>end-time</i> and duration is infinite.</p>
Step 6	<p>send-lifetime <i>start-time</i> {infinite <i>end-time</i> duration <i>seconds</i>}</p> <p>Example:</p> <pre>Device(config-keychain)# accept-lifetime 23:30:00 Jan 25 1019 infinite</pre>	<p>(Optional) Specifies the time period during which the key can be sent.</p> <p>The <i>start-time</i> and <i>end-time</i> syntax can be either <i>hh:mm:ss Month date year</i> or <i>hh:mm:ss date Month year</i>. The default is forever with the default <i>start-time</i> and the earliest acceptable date as January 1, 1993. The default <i>end-time</i> and duration is infinite.</p>
Step 7	<p>end</p> <p>Example:</p> <pre>Device(config-keychain)# end</pre>	Returns to privileged EXEC mode.
Step 8	<p>show key chain</p> <p>Example:</p> <pre>Device# show key chain</pre>	Displays authentication key information.
Step 9	<p>copy running-config startup-config</p> <p>Example:</p> <pre>Device# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Monitoring and Maintaining the IP Network

You can remove all contents of a particular cache, table, or database. You can also display specific statistics.

Table 98: Commands to Clear IP Routes or Display Route Status

Command	Purpose
clear ip route { <i>network</i> [<i>mask</i> *]}	Clears one or more routes from the IP routing table.

Command	Purpose
show ip protocols	Displays the parameters and state of the active routing protocol process.
show ip route [<i>address</i> [<i>mask</i>] [longer-prefixes]] [<i>protocol</i> [<i>process-id</i>]]	Displays the current state of the routing table.
show ip route summary	Displays the current state of the routing table in summary form.
show ip route supernets-only	Displays supernets.
show ip cache	Displays the routing table used to switch IP traffic.
show route-map [<i>map-name</i>]	Displays all route maps configured or only the one specified.



CHAPTER 43

Configuring Bidirectional Forwarding Detection

- [Prerequisites for Bidirectional Forwarding Detection, on page 923](#)
- [Restrictions for Bidirectional Forwarding Detection, on page 923](#)
- [Information About Bidirectional Forwarding Detection, on page 924](#)
- [How to Configure Bidirectional Forwarding Detection, on page 927](#)
- [Configuration Examples for Bidirectional Forwarding Detection, on page 932](#)
- [Additional References for Bidirectional Forwarding Detection, on page 933](#)
- [Feature Information for Bidirectional Forwarding Detection, on page 934](#)

Prerequisites for Bidirectional Forwarding Detection

Prerequisites for BFD include:

- The switch's feature set is IP Base or higher. The IP Base feature set supports only Enhanced Interior Gateway Routing Protocol (EIGRP) stub routing, without BFD. The IP service feature set supports EIGRP with BFD.
- IP routing must be enabled on all participating switches
- Before BFD is deployed, configure one of the IP routing protocols supported by BFD on the switches. Also, implement fast convergence for the routing protocol that you plan to use.

Restrictions for Bidirectional Forwarding Detection

Restrictions for BFD include:

- BFD works only for directly connected neighbors. BFD neighbors must be no more than one IP hop away. Multihop configurations are not supported.
- The switch supports up to 100 BFD sessions with a minimum hello interval of 100 ms and a multiplier of 3. The multiplier specifies the minimum number of consecutive packets that can be missed before a session is declared down.
- To enable echo mode the peer system must be configured with the `no ip redirects` command.

Information About Bidirectional Forwarding Detection

BFD Operation

BFD provides a low-overhead, short-duration method of detecting failures in the forwarding path between two adjacent routers, including the interfaces, data links, and forwarding planes.

BFD is a detection protocol that you enable at the interface and routing protocol levels. Cisco supports the BFD asynchronous mode, which depends on the sending of BFD control packets between two systems to activate and maintain BFD neighbor sessions between routers. Therefore, in order for a BFD session to be created, you must configure BFD on both systems (or BFD peers). Once BFD has been enabled on the interfaces and at the router level for the appropriate routing protocols, a BFD session is created, BFD timers are negotiated, and the BFD peers will begin to send BFD control packets to each other at the negotiated interval.

Cisco supports BFD echo mode. Echo packets are sent by the forwarding engine and are forwarded back along the same path to perform detection. The BFD session at the other end does not participate in the actual forwarding of the echo packets.

This section includes the following subsections:

BFD Version Interoperability

The switch supports BFD Version 1 as well as BFD Version 0. All BFD sessions come up as Version 1 by default and will be interoperable with Version 0. The system automatically performs BFD version detection, and BFD sessions between neighbors will run in the highest common BFD version between neighbors. For example, if one BFD neighbor is running BFD Version 0 and the other BFD neighbor is running Version 1, the session will run BFD Version 0. The output from the **show bfd neighbors [details]** command will verify which BFD version a BFD neighbor is running.

BFD Session Limits

The minimum number of BFD sessions that can be created varies with the “hello” interval. With “hello” intervals of 100ms, 100 sessions are permitted. More sessions are permitted at larger hello intervals. For a VLAN interface, the minimum “hello” interval is 600ms.

BFD Support for Nonbroadcast Media Interfaces

The BFD feature is supported on VLAN interfaces on the switch.

The **bfd interval** command must be configured on the interface to initiate BFD monitoring.

BFD Support for Nonstop Forwarding with Stateful Switchover

Typically, when a networking device restarts, all routing peers of that device detect that the device went down and then came back up. This transition results in a routing flap, which could spread across multiple routing domains. Routing flaps caused by routing restarts create routing instabilities, which are detrimental to the overall network performance. Nonstop forwarding (NSF) helps to suppress routing flaps in devices that are enabled with stateful switchover (SSO), thereby reducing network instability.

NSF allows for the forwarding of data packets to continue along known routes while the routing protocol information is being restored after a switchover. With NSF, peer networking devices do not experience routing flaps. Data traffic is forwarded through intelligent line cards or dual forwarding processors while the standby RP assumes control from the failed active RP during a switchover. The ability of line cards and forwarding processors to remain up through a switchover and to be kept current with the Forwarding Information Base (FIB) on the active RP is key to NSF operation.

In devices that support dual RPs, SSO establishes one of the RPs as the active processor; the other RP is designated as the standby processor, and then synchronizes information between them. A switchover from the active to the standby processor occurs when the active RP fails, when it is removed from the networking device, or when it is manually taken down for maintenance.

BFD Support for Stateful Switchover

The BFD protocol provides short-duration detection of failures in the path between adjacent forwarding engines. In network deployments that use dual RP switches (to provide redundancy), the switches have a graceful restart mechanism that protects the forwarding state during a switchover between the active RP and the standby RP.

Stateful BFD on the Standby RP

To ensure a successful switchover to the standby RP, the BFD protocol uses checkpoint messages to send session information from the active RP Cisco IOS instance to the standby RP Cisco IOS instance. The session information includes local and remote discriminators, adjacent router timer information, BFD setup information, and session-specific information such as the type of session and the session version. In addition, the BFD protocol sends session creation and deletion checkpoint messages to create or delete a session on the standby RP.

The BFD sessions on the standby RP do not receive or send packets and do not process expired timers. These sessions wait for a switchover to occur and then send packets for any active sessions so that sessions do not time out on adjacent switches.

When the BFD protocol on the standby RP is notified of a switchover it changes its state to active, registers itself with Cisco Express Forwarding so that it can receive packets, and then sends packets for any elements that have expired.

BFD also uses checkpoint messages to ensure that sessions created by clients on the active RP are maintained during a switchover. When a switchover occurs, BFD starts an SSO reclaim timer. Clients must reclaim their sessions within the duration specified by the reclaim timer or else the session is deleted.

Timer values are different based on the number of BFD sessions and the platform.

Table 99: BFD Timer Values on the switch

Maximum Number of BFD Sessions	BFD Session Type	Minimum Timer Value (ms)	Clients	Comments
100	Async/echo	100 multiplier 3	All	A multiple of 5 is recommended for SSO switches.

BFD Support for Static Routing

Unlike dynamic routing protocols, such as OSPF and BGP, static routing has no method of peer discovery. Therefore, when BFD is configured, the reachability of the gateway is completely dependent on the state of the BFD session to the specified neighbor. Unless the BFD session is up, the gateway for the static route is considered unreachable, and therefore the affected routes will not be installed in the appropriate Routing Information Base (RIB).

For a BFD session to be successfully established, BFD must be configured on the interface on the peer and there must be a BFD client registered on the peer for the address of the BFD neighbor. When an interface is used by dynamic routing protocols, the latter requirement is usually met by configuring the routing protocol instances on each neighbor for BFD. When an interface is used exclusively for static routing, this requirement must be met by configuring static routes on the peers.

If a BFD configuration is removed from the remote peer while the BFD session is in the up state, the updated state of the BFD session is not signaled to the static static. This will cause the static route to remain in the RIB. The only workaround is to remove the IPv4 static BFD neighbor configuration so that the static route no longer tracks BFD session state.

Configuring BFD Echo Mode

BFD echo mode is enabled by default, but you can disable it such that it can run independently in each direction.

BFD echo mode works with asynchronous BFD. Echo packets are sent by the forwarding engine and forwarded back along the same path in order to perform detection--the BFD session at the other end does not participate in the actual forwarding of the echo packets. The echo function and the forwarding engine are responsible for the detection process; therefore, the number of BFD control packets that are sent out between two BFD neighbors is reduced. In addition, because the forwarding engine is testing the forwarding path on the remote (neighbor) system without involving the remote system, there is an opportunity to improve the interpacket delay variance, thereby achieving quicker failure detection times than when using BFD Version 0 with BFD control packets for the BFD session.

Echo mode is described as without asymmetry when it is running on both sides (both BFD neighbors are running echo mode).

Prerequisites

BFD must be running on all participating switches.

Before using BFD echo mode, you must disable the sending of Internet Control Message Protocol (ICMP) redirect messages by entering the **no ip redirects** command, in order to avoid high CPU utilization.

The baseline parameters for BFD sessions on the interfaces over which you want to run BFD sessions to BFD neighbors must be configured. See the Configuring BFD Session Parameters on the Interface section for more information.

Restrictions

BFD echo mode, which is supported in BFD Version 1.



Note BFD echo mode does not work in conjunction with Unicast Reverse Path Forwarding (uRPF) configuration. If BFD echo mode and uRPF configurations are enabled, then the sessions will flap.

How to Configure Bidirectional Forwarding Detection

You start a BFD process by configuring BFD on the interface. When the BFD process is started, no entries are created in the adjacency database; in other words, no BFD control packets are sent or received. BFD echo mode, which is supported in BFD Version 1.

BFD echo packets are sent and received, in addition to BFD control packets. The adjacency creation takes place once you have configured BFD support for the applicable routing protocols. This section contains the following procedures:

Configuring BFD Session Parameters on the Interface

Perform this task to configure BFD on an interface by setting the baseline BFD session parameters on the interface. Repeat this task on each interface over which you want to run BFD sessions to BFD neighbors.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *type number*
4. **bfd interval** *milliseconds min_rx milliseconds multiplier interval-multiplier*
5. **no bfd echo**
6. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	interface <i>type number</i> Example: Switch(config)# interface GigabitEthernet 6/1	Specifies an interface type and number, and places the device in interface configuration mode.
Step 4	bfd interval <i>milliseconds min_rx milliseconds multiplier interval-multiplier</i> Example: Switch(config-if)# no bfd echo	Enables BFD on the interface. Disables BFD echo mode to enable Hardware Off-load.

	Command or Action	Purpose
Step 5	no bfd echo Example: Switch(config-if)# no bfd echo	Disables BFD echo mode to enable Hardware Off-load.
Step 6	end Example: Switch(config-if)# end	Exits interface configuration mode and returns to privileged EXEC mode.

Configuring BFD Support for Static Routing

Perform this task to configure BFD support for static routing. Repeat the steps in this procedure on each BFD neighbor. For more information, see the “Example: Configuring BFD Support for Static Routing” section

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *type number*
4. **no switchport**
5. **ip address** *ip-address mask*
6. **bfd interval** *milliseconds min_rx milliseconds multiplier interval-multiplier*
7. **exit**
8. **ip route static bfd** *interface-type interface-number ip-address* [**group** *group-name* [**passive**]]
9. **ip route** [**vrf** *vrf-name*] *prefix mask* {*ip-address* | *interface-type interface-number* [*ip-address*]} [**dhcp**] [*distance*] [**name** *next-hop-name*] [**permanent** | **track** *number*] [**tag** *tag*]
10. **exit**
11. **show ip static route**
12. **show ip static route bfd**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.

	Command or Action	Purpose
Step 3	interface <i>type number</i> Example: <pre>Switch(config)# interface gigabitethernet 6/1</pre>	Configures an interface and enters interface configuration mode.
Step 4	no switchport Example: <pre>Switch(config)# no switchport</pre>	Changes the interface to Layer 3.
Step 5	ip address <i>ip-address mask</i> Example: <pre>Switch(config-if)# ip address 10.201.201.1 255.255.255.0</pre>	Configures an IP address for the interface.
Step 6	bfd interval <i>milliseconds min_rx milliseconds multiplier interval-multiplier</i> Example: <pre>Switch(config-if)# bfd interval 500 min_rx 500 multiplier 5</pre>	Enables BFD on the interface.
Step 7	exit Example: <pre>Switch(config-if)# exit</pre>	Exits interface configuration mode and returns to global configuration mode.
Step 8	ip route static bfd <i>interface-type interface-number ip-address [group group-name [passive]]</i> Example: <pre>Switch(config)# ip route static bfd serial 2/0 10.1.1.1 group group1 passive</pre>	Specifies a static route BFD neighbor. <ul style="list-style-type: none"> The <i>interface-type</i>, <i>interface-number</i>, and <i>ip-address</i> arguments are required because BFD support exists only for directly connected neighbors.
Step 9	ip route [vrf <i>vrf-name</i>] <i>prefix mask {ip-address interface-type interface-number [ip-address]}</i> [dhcp] [<i>distance</i>] [name <i>next-hop-name</i>] [permanent track number] [tag <i>tag</i>] Example: <pre>Switch(config)# ip route 10.0.0.0 255.0.0.0 Gigabitethernet 6/1 10.201.201.2</pre>	Specifies a static route BFD neighbor.
Step 10	exit Example: <pre>Switch(config)# exit</pre>	Exits global configuration mode and returns to privileged EXEC mode.

	Command or Action	Purpose
Step 11	show ip static route Example: <pre>Switch# show ip static route</pre>	(Optional) Displays static route database information.
Step 12	show ip static route bfd Example: <pre>Switch# show ip static route bfd</pre>	(Optional) Displays information about the static BFD configuration from the configured BFD groups and non-group entries.

Configuring the BFD Slow Timer

This task shows how to change the value of the BFD slow timer. Repeat the steps in this task for each BFD switch.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **bfd slow-timer** *milliseconds*
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Switch> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 3	bfd slow-timer <i>milliseconds</i> Example: <pre>Switch(config)# bfd slow-timer 12000</pre>	Configures the BFD slow timer.
Step 4	end Example: <pre>Switch(config)# end</pre>	Exits global configuration mode and returns to privileged EXEC mode.

Disabling BFD Echo Mode Without Asymmetry

This task shows how to disable BFD echo mode without asymmetry—no echo packets will be sent by the switch, and the switch will not forward BFD echo packets that are received from any neighbor switches.

Repeat the steps in this task for each BFD switch.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **no bfd echo**
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Switch> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 3	no bfd echo Example: <pre>Switch(config)# no bfd echo</pre>	Disables BFD echo mode. <ul style="list-style-type: none"> • Use the no form to disable BFD echo mode.
Step 4	end Example: <pre>Switch(config)# end</pre>	Exits global configuration mode and returns to privileged EXEC mode.

Monitoring and Troubleshooting BFD

This section describes how to retrieve BFD information for maintenance and troubleshooting. The commands in these tasks can be entered as needed, in any order.

To monitor and troubleshoot BFD, perform the following steps:

SUMMARY STEPS

1. **enable**
2. **show bfd neighbors [details]**
3. **debug bfd [packet | event]**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	show bfd neighbors [details] Example: Switch# show bfd neighbors details	(Optional) Displays the BFD adjacency database. <ul style="list-style-type: none"> • The details keyword shows all BFD protocol parameters and timers per neighbor.
Step 3	debug bfd [packet event] Example: Switch# debug bfd packet	(Optional) Displays debugging information about BFD packets.

Configuration Examples for Bidirectional Forwarding Detection

Example: Configuring BFD Session Parameters on the Interface

```
Switch# configure terminal
Switch(config)# interface GigabitEthernet 6/1
Switch(config-if)# bfd interval 50 min_rx 50 multiplier 5
Switch(config-if)# no bfd echo
```

Example: Configuring BFD Support for Static Routing

In the following example, the network consists of Device A and Device B. Serial interface 2/0 on Device A is connected to the same network as serial interface 2/0 on Device B. In order for the BFD session to come up, Device B must be configured.

Device A

```
configure terminal
interface Serial 2/0
ip address 10.201.201.1 255.255.255.0
bfd interval 500 min_rx 500 multiplier 5
ip route static bfd Serial 2/0 10.201.201.2
ip route 10.0.0.0 255.0.0.0 Serial 2/0 10.201.201.2
```

Device B

```
configure terminal
interface Serial 2/0
ip address 10.201.201.2 255.255.255.0
```

```
bfd interval 500 min_rx 500 multiplier 5
ip route static bfd Serial 2/0 10.201.201.1
ip route 10.1.1.1 255.255.255.255 Serial 2/0 10.201.201.1
```

Note that the static route on Device B exists solely to enable the BFD session between 10.201.201.1 and 10.201.201.2. If there is no useful static route that needs to be configured, select a prefix that will not affect packet forwarding, for example, the address of a locally configured loopback interface.

In the following example, there is an active static BFD configuration to reach 209.165.200.225 through Ethernet interface 0/0 in the BFD group testgroup. As soon as the static route is configured that is tracked by the configured static BFD, a single hop BFD session is initiated to 209.165.200.225 through Ethernet interface 0/0. The prefix 10.0.0.0/8 is added to the RIB if a BFD session is successfully established.

```
configure terminal
ip route static bfd Ethernet 0/0 209.165.200.225 group testgroup
ip route 10.0.0.0 255.255.255.224 Ethernet 0/0 209.165.200.225
```

In the following example, a BFD session to 209.165.200.226 through Ethernet interface 0/0.1001 is marked to use the group testgroup. That is, this configuration is a passive static BFD. Though there are static routes to be tracked by the second static BFD configuration, a BFD session is not triggered for 209.165.200.226 through Ethernet interface 0/0.1001. The existence of the prefixes 10.1.1.1/8 and 10.2.2.2/8 is controlled by the active static BFD session (Ethernet interface 0/0 209.165.200.225).

```
configure terminal
ip route static bfd Ethernet 0/0 209.165.200.225 group testgroup
ip route 10.0.0.0 255.255.255.224 Ethernet 0/0 209.165.200.225
ip route static bfd Ethernet 0/0.1001 209.165.200.226 group testgroup passive
ip route 10.1.1.1 255.255.255.224 Ethernet 0/0.1001 209.165.200.226
ip route 10.2.2.2 255.255.255.224 Ethernet 0/0.1001 209.165.200.226
```

Example: Configuring BFD Slow Timer

```
Switch# configure terminal
Switch(config)# bfd slow-timer 12000
```

Additional References for Bidirectional Forwarding Detection

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies.	http://www.cisco.com/support
To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds.	
Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	

Feature Information for Bidirectional Forwarding Detection

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Table 100: Feature Information for Bidirectional Forwarding Detection

Feature Name	Releases	Feature Information
Bidirectional Forwarding Detection	Cisco IOS 15.2(4)E1	<p>The Bidirectional Forwarding Detection (BFD) protocol is a detection protocol that is designed to provide fast forwarding path failure detection for all media types, encapsulations, topologies, and routing protocols.</p> <p>BFD provides a consistent failure detection method for network administrators, in addition to fast forwarding path failure detection. Because the network administrator can use BFD to detect forwarding path failures at a uniform rate, rather than the variable rates for different routing protocol hello mechanisms, network profiling and planning will be easier, and reconvergence time will be consistent and predictable.</p> <p>This feature was implemented on Cisco Catalyst 2960-XR Series Switches.</p>



CHAPTER 44

Bidirectional Forwarding Detection

The Bidirectional Forwarding Detection (BFD) protocol is a detection protocol that provides fast forwarding path failure detection for all media types, encapsulations, topologies, and routing protocols.

BFD provides a consistent failure detection method for network administrators, in addition to fast forwarding path failure detection. Because the network administrator can use BFD to detect forwarding path failures at a uniform rate, rather than the variable rates for different routing protocol hello mechanisms, network profiling and planning will be easier, and reconvergence time will be consistent and predictable.

This module explains how to configure multihop BFD sessions.

- [Prerequisites for Bidirectional Forwarding Detection, on page 935](#)
- [Restrictions for Bidirectional Forwarding Detection, on page 935](#)
- [Information About Bidirectional Forwarding Detection, on page 936](#)
- [How to Configure Bidirectional Forwarding Detection, on page 939](#)
- [Configuration Examples for Bidirectional Forwarding Detection, on page 944](#)
- [Additional References for Bidirectional Forwarding Detection, on page 946](#)
- [Feature Information for Bidirectional Forwarding Detection, on page 946](#)

Prerequisites for Bidirectional Forwarding Detection

Prerequisites for BFD include:

- The switch's feature set is IP Base or higher. The IP Base feature set supports only Enhanced Interior Gateway Routing Protocol (EIGRP) stub routing, without BFD. The IP service feature set supports EIGRP with BFD.
- IP routing must be enabled on all participating switches
- Before BFD is deployed, configure one of the IP routing protocols supported by BFD on the switches. Also, implement fast convergence for the routing protocol that you plan to use.

Restrictions for Bidirectional Forwarding Detection

Restrictions for BFD include:

- BFD works only for directly connected neighbors. BFD neighbors must be no more than one IP hop away. Multihop configurations are not supported.

- The switch supports up to 100 BFD sessions with a minimum hello interval of 100 ms and a multiplier of 3. The multiplier specifies the minimum number of consecutive packets that can be missed before a session is declared down.
- To enable echo mode the peer system must be configured with the `no ip redirects` command.

Information About Bidirectional Forwarding Detection

BFD Operation

BFD provides a low-overhead, short-duration method of detecting failures in the forwarding path between two adjacent routers, including the interfaces, data links, and forwarding planes.

BFD is a detection protocol that you enable at the interface and routing protocol levels. Cisco supports the BFD asynchronous mode, which depends on the sending of BFD control packets between two systems to activate and maintain BFD neighbor sessions between routers. Therefore, in order for a BFD session to be created, you must configure BFD on both systems (or BFD peers). Once BFD has been enabled on the interfaces and at the router level for the appropriate routing protocols, a BFD session is created, BFD timers are negotiated, and the BFD peers will begin to send BFD control packets to each other at the negotiated interval.

Cisco supports BFD echo mode. Echo packets are sent by the forwarding engine and are forwarded back along the same path to perform detection. The BFD session at the other end does not participate in the actual forwarding of the echo packets.

This section includes the following subsections:

BFD Version Interoperability

The switch supports BFD Version 1 as well as BFD Version 0. All BFD sessions come up as Version 1 by default and will be interoperable with Version 0. The system automatically performs BFD version detection, and BFD sessions between neighbors will run in the highest common BFD version between neighbors. For example, if one BFD neighbor is running BFD Version 0 and the other BFD neighbor is running Version 1, the session will run BFD Version 0. The output from the `show bfd neighbors [details]` command will verify which BFD version a BFD neighbor is running.

BFD Session Limits

The minimum number of BFD sessions that can be created varies with the “hello” interval. With “hello” intervals of 100ms, 100 sessions are permitted. More sessions are permitted at larger hello intervals. For a VLAN interface, the minimum “hello” interval is 600ms.

BFD Support for Nonbroadcast Media Interfaces

The BFD feature is supported on VLAN interfaces on the switch.

The `bfd interval` command must be configured on the interface to initiate BFD monitoring.

BFD Support for Nonstop Forwarding with Stateful Switchover

Typically, when a networking device restarts, all routing peers of that device detect that the device went down and then came back up. This transition results in a routing flap, which could spread across multiple routing domains. Routing flaps caused by routing restarts create routing instabilities, which are detrimental to the overall network performance. Nonstop forwarding (NSF) helps to suppress routing flaps in devices that are enabled with stateful switchover (SSO), thereby reducing network instability.

NSF allows for the forwarding of data packets to continue along known routes while the routing protocol information is being restored after a switchover. With NSF, peer networking devices do not experience routing flaps. Data traffic is forwarded through intelligent line cards or dual forwarding processors while the standby RP assumes control from the failed active RP during a switchover. The ability of line cards and forwarding processors to remain up through a switchover and to be kept current with the Forwarding Information Base (FIB) on the active RP is key to NSF operation.

In devices that support dual RPs, SSO establishes one of the RPs as the active processor; the other RP is designated as the standby processor, and then synchronizes information between them. A switchover from the active to the standby processor occurs when the active RP fails, when it is removed from the networking device, or when it is manually taken down for maintenance.

BFD Support for Stateful Switchover

The BFD protocol provides short-duration detection of failures in the path between adjacent forwarding engines. In network deployments that use dual RP switches (to provide redundancy), the switches have a graceful restart mechanism that protects the forwarding state during a switchover between the active RP and the standby RP.

Stateful BFD on the Standby RP

To ensure a successful switchover to the standby RP, the BFD protocol uses checkpoint messages to send session information from the active RP Cisco IOS instance to the standby RP Cisco IOS instance. The session information includes local and remote discriminators, adjacent router timer information, BFD setup information, and session-specific information such as the type of session and the session version. In addition, the BFD protocol sends session creation and deletion checkpoint messages to create or delete a session on the standby RP.

The BFD sessions on the standby RP do not receive or send packets and do not process expired timers. These sessions wait for a switchover to occur and then send packets for any active sessions so that sessions do not time out on adjacent switches.

When the BFD protocol on the standby RP is notified of a switchover it changes its state to active, registers itself with Cisco Express Forwarding so that it can receive packets, and then sends packets for any elements that have expired.

BFD also uses checkpoint messages to ensure that sessions created by clients on the active RP are maintained during a switchover. When a switchover occurs, BFD starts an SSO reclaim timer. Clients must reclaim their sessions within the duration specified by the reclaim timer or else the session is deleted.

Timer values are different based on the number of BFD sessions and the platform.

Table 101: BFD Timer Values on the switch

Maximum Number of BFD Sessions	BFD Session Type	Minimum Timer Value (ms)	Clients	Comments
100	Async/echo	100 multiplier 3	All	A multiple of 5 is recommended for SSO switches.

BFD Support for Static Routing

Unlike dynamic routing protocols, such as OSPF and BGP, static routing has no method of peer discovery. Therefore, when BFD is configured, the reachability of the gateway is completely dependent on the state of the BFD session to the specified neighbor. Unless the BFD session is up, the gateway for the static route is considered unreachable, and therefore the affected routes will not be installed in the appropriate Routing Information Base (RIB).

For a BFD session to be successfully established, BFD must be configured on the interface on the peer and there must be a BFD client registered on the peer for the address of the BFD neighbor. When an interface is used by dynamic routing protocols, the latter requirement is usually met by configuring the routing protocol instances on each neighbor for BFD. When an interface is used exclusively for static routing, this requirement must be met by configuring static routes on the peers.

If a BFD configuration is removed from the remote peer while the BFD session is in the up state, the updated state of the BFD session is not signaled to the static static. This will cause the static route to remain in the RIB. The only workaround is to remove the IPv4 static BFD neighbor configuration so that the static route no longer tracks BFD session state.

Configuring BFD Echo Mode

BFD echo mode is enabled by default, but you can disable it such that it can run independently in each direction.

BFD echo mode works with asynchronous BFD. Echo packets are sent by the forwarding engine and forwarded back along the same path in order to perform detection--the BFD session at the other end does not participate in the actual forwarding of the echo packets. The echo function and the forwarding engine are responsible for the detection process; therefore, the number of BFD control packets that are sent out between two BFD neighbors is reduced. In addition, because the forwarding engine is testing the forwarding path on the remote (neighbor) system without involving the remote system, there is an opportunity to improve the interpacket delay variance, thereby achieving quicker failure detection times than when using BFD Version 0 with BFD control packets for the BFD session.

Echo mode is described as without asymmetry when it is running on both sides (both BFD neighbors are running echo mode).

Prerequisites

BFD must be running on all participating switches.

Before using BFD echo mode, you must disable the sending of Internet Control Message Protocol (ICMP) redirect messages by entering the **no ip redirects** command, in order to avoid high CPU utilization.

The baseline parameters for BFD sessions on the interfaces over which you want to run BFD sessions to BFD neighbors must be configured. See the Configuring BFD Session Parameters on the Interface section for more information.

Restrictions

BFD echo mode, which is supported in BFD Version 1.



Note BFD echo mode does not work in conjunction with Unicast Reverse Path Forwarding (uRPF) configuration. If BFD echo mode and uRPF configurations are enabled, then the sessions will flap.

How to Configure Bidirectional Forwarding Detection

You start a BFD process by configuring BFD on the interface. When the BFD process is started, no entries are created in the adjacency database; in other words, no BFD control packets are sent or received. BFD echo mode, which is supported in BFD Version 1.

BFD echo packets are sent and received, in addition to BFD control packets. The adjacency creation takes place once you have configured BFD support for the applicable routing protocols. This section contains the following procedures:

Configuring BFD Session Parameters on the Interface

Perform this task to configure BFD on an interface by setting the baseline BFD session parameters on the interface. Repeat this task on each interface over which you want to run BFD sessions to BFD neighbors.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *type number*
4. **bfd interval** *milliseconds* **min_rx** *milliseconds* **multiplier** *interval-multiplier*
5. **no bfd echo**
6. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.

	Command or Action	Purpose
Step 3	interface <i>type number</i> Example: Switch(config)# interface GigabitEthernet 6/1	Specifies an interface type and number, and places the device in interface configuration mode.
Step 4	bfd interval <i>milliseconds min_rx milliseconds multiplier interval-multiplier</i> Example: Switch(config-if)# no bfd echo	Enables BFD on the interface. Disables BFD echo mode to enable Hardware Off-load.
Step 5	no bfd echo Example: Switch(config-if)# no bfd echo	Disables BFD echo mode to enable Hardware Off-load.
Step 6	end Example: Switch(config-if)# end	Exits interface configuration mode and returns to privileged EXEC mode.

Configuring BFD Support for Static Routing

Perform this task to configure BFD support for static routing. Repeat the steps in this procedure on each BFD neighbor. For more information, see the “Example: Configuring BFD Support for Static Routing” section

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *type number*
4. **no switchport**
5. **ip address** *ip-address mask*
6. **bfd interval** *milliseconds min_rx milliseconds multiplier interval-multiplier*
7. **exit**
8. **ip route static bfd** *interface-type interface-number ip-address* [**group** *group-name* [**passive**]]
9. **ip route** [**vrf** *vrf-name*] *prefix mask* {*ip-address* | *interface-type interface-number* [*ip-address*]} [**dhcp**] [*distance*] [**name** *next-hop-name*] [**permanent** | **track** *number*] [**tag** *tag*]
10. **exit**
11. **show ip static route**
12. **show ip static route bfd**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	interface <i>type number</i> Example: Switch(config)# interface gigabitethernet 6/1	Configures an interface and enters interface configuration mode.
Step 4	no switchport Example: Switch(config)# no switchport	Changes the interface to Layer 3.
Step 5	ip address <i>ip-address mask</i> Example: Switch(config-if)# ip address 10.201.201.1 255.255.255.0	Configures an IP address for the interface.
Step 6	bfd interval <i>milliseconds min_rx milliseconds multiplier interval-multiplier</i> Example: Switch(config-if)# bfd interval 500 min_rx 500 multiplier 5	Enables BFD on the interface.
Step 7	exit Example: Switch(config-if)# exit	Exits interface configuration mode and returns to global configuration mode.
Step 8	ip route static bfd <i>interface-type interface-number ip-address [group group-name [passive]]</i> Example: Switch(config)# ip route static bfd serial 2/0 10.1.1.1 group group1 passive	Specifies a static route BFD neighbor. <ul style="list-style-type: none"> • The <i>interface-type</i>, <i>interface-number</i>, and <i>ip-address</i> arguments are required because BFD support exists only for directly connected neighbors.
Step 9	ip route [vrf <i>vrf-name</i>] <i>prefix mask {ip-address interface-type interface-number [ip-address]}</i> [dhcp]	Specifies a static route BFD neighbor.

	Command or Action	Purpose
	<p>[<i>distance</i>] [<i>name next-hop-name</i>] [permanent track number] [tag tag]</p> <p>Example:</p> <pre>Switch(config)# ip route 10.0.0.0 255.0.0.0 GigabitEthernet 6/1 10.201.201.2</pre>	
Step 10	<p>exit</p> <p>Example:</p> <pre>Switch(config)# exit</pre>	Exits global configuration mode and returns to privileged EXEC mode.
Step 11	<p>show ip static route</p> <p>Example:</p> <pre>Switch# show ip static route</pre>	(Optional) Displays static route database information.
Step 12	<p>show ip static route bfd</p> <p>Example:</p> <pre>Switch# show ip static route bfd</pre>	(Optional) Displays information about the static BFD configuration from the configured BFD groups and non-group entries.

Configuring the BFD Slow Timer

This task show how to change the value of the BFD slow timer. Repeat the steps in this task for each BFD switch.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **bfd slow-timer *milliseconds***
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p>enable</p> <p>Example:</p> <pre>Switch> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	<p>configure terminal</p> <p>Example:</p> <pre>Switch# configure terminal</pre>	Enters global configuration mode.

	Command or Action	Purpose
Step 3	bfd slow-timer <i>milliseconds</i> Example: Switch(config)# bfd slow-timer 12000	Configures the BFD slow timer.
Step 4	end Example: Switch(config)# end	Exits global configuration mode and returns to privileged EXEC mode.

Disabling BFD Echo Mode Without Asymmetry

This task shows how to disable BFD echo mode without asymmetry—no echo packets will be sent by the switch, and the switch will not forward BFD echo packets that are received from any neighbor switches.

Repeat the steps in this task for each BFD switch.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **no bfd echo**
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	no bfd echo Example: Switch(config)# no bfd echo	Disables BFD echo mode. <ul style="list-style-type: none"> • Use the no form to disable BFD echo mode.
Step 4	end Example: Switch(config)# end	Exits global configuration mode and returns to privileged EXEC mode.

Monitoring and Troubleshooting BFD

This section describes how to retrieve BFD information for maintenance and troubleshooting. The commands in these tasks can be entered as needed, in any order.

To monitor and troubleshoot BFD, perform the following steps:

SUMMARY STEPS

1. **enable**
2. **show bfd neighbors [details]**
3. **debug bfd [packet | event]**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	show bfd neighbors [details] Example: Switch# show bfd neighbors details	(Optional) Displays the BFD adjacency database. <ul style="list-style-type: none"> • The details keyword shows all BFD protocol parameters and timers per neighbor.
Step 3	debug bfd [packet event] Example: Switch# debug bfd packet	(Optional) Displays debugging information about BFD packets.

Configuration Examples for Bidirectional Forwarding Detection

Example: Configuring BFD Session Parameters on the Interface

```
Switch# configure terminal
Switch(config)# interface GigabitEthernet 6/1
Switch(config-if)# bfd interval 50 min_rx 50 multiplier 5
Switch(config-if)# no bfd echo
```

Example: Configuring BFD Support for Static Routing

In the following example, the network consists of Device A and Device B. Serial interface 2/0 on Device A is connected to the same network as serial interface 2/0 on Device B. In order for the BFD session to come up, Device B must be configured.

Device A

```
configure terminal
interface Serial 2/0
ip address 10.201.201.1 255.255.255.0
bfd interval 500 min_rx 500 multiplier 5
ip route static bfd Serial 2/0 10.201.201.2
ip route 10.0.0.0 255.0.0.0 Serial 2/0 10.201.201.2
```

Device B

```
configure terminal
interface Serial 2/0
ip address 10.201.201.2 255.255.255.0
bfd interval 500 min_rx 500 multiplier 5
ip route static bfd Serial 2/0 10.201.201.1
ip route 10.1.1.1 255.255.255.255 Serial 2/0 10.201.201.1
```

Note that the static route on Device B exists solely to enable the BFD session between 10.201.201.1 and 10.201.201.2. If there is no useful static route that needs to be configured, select a prefix that will not affect packet forwarding, for example, the address of a locally configured loopback interface.

In the following example, there is an active static BFD configuration to reach 209.165.200.225 through Ethernet interface 0/0 in the BFD group testgroup. As soon as the static route is configured that is tracked by the configured static BFD, a single hop BFD session is initiated to 209.165.200.225 through Ethernet interface 0/0. The prefix 10.0.0.0/8 is added to the RIB if a BFD session is successfully established.

```
configure terminal
ip route static bfd Ethernet 0/0 209.165.200.225 group testgroup
ip route 10.0.0.0 255.255.255.224 Ethernet 0/0 209.165.200.225
```

In the following example, a BFD session to 209.165.200.226 through Ethernet interface 0/0.1001 is marked to use the group testgroup. That is, this configuration is a passive static BFD. Though there are static routes to be tracked by the second static BFD configuration, a BFD session is not triggered for 209.165.200.226 through Ethernet interface 0/0.1001. The existence of the prefixes 10.1.1.1/8 and 10.2.2.2/8 is controlled by the active static BFD session (Ethernet interface 0/0 209.165.200.225).

```
configure terminal
ip route static bfd Ethernet 0/0 209.165.200.225 group testgroup
ip route 10.0.0.0 255.255.255.224 Ethernet 0/0 209.165.200.225
ip route static bfd Ethernet 0/0.1001 209.165.200.226 group testgroup passive
ip route 10.1.1.1 255.255.255.224 Ethernet 0/0.1001 209.165.200.226
ip route 10.2.2.2 255.255.255.224 Ethernet 0/0.1001 209.165.200.226
```

Example: Configuring BFD Slow Timer

```
Switch# configure terminal
Switch(config)# bfd slow-timer 12000
```

Additional References for Bidirectional Forwarding Detection

Technical Assistance

Description	Link
<p>The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies.</p> <p>To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds.</p> <p>Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.</p>	http://www.cisco.com/support

Feature Information for Bidirectional Forwarding Detection

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Table 102: Feature Information for Bidirectional Forwarding Detection

Feature Name	Releases	Feature Information
Bidirectional Forwarding Detection	Cisco IOS 15.2(4)E1	<p>The Bidirectional Forwarding Detection (BFD) protocol is a detection protocol that is designed to provide fast forwarding path failure detection for all media types, encapsulations, topologies, and routing protocols.</p> <p>BFD provides a consistent failure detection method for network administrators, in addition to fast forwarding path failure detection. Because the network administrator can use BFD to detect forwarding path failures at a uniform rate, rather than the variable rates for different routing protocol hello mechanisms, network profiling and planning will be easier, and reconvergence time will be consistent and predictable.</p> <p>This feature was implemented on Cisco Catalyst 2960-XR Series Switches.</p>



CHAPTER 45

Configuring IPv6 First Hop Security

- Finding Feature Information, on page 947
- Prerequisites for First Hop Security in IPv6, on page 947
- Restrictions for First Hop Security in IPv6, on page 948
- Information about First Hop Security in IPv6, on page 948
- How to Configure an IPv6 Snooping Policy, on page 951
- **How to Configure the IPv6 Binding Table Content** , on page 955
- How to Configure an IPv6 Neighbor Discovery Inspection Policy, on page 956
- How to Attach an IPv6 Neighbor Discovery Multicast Suppress Policy on a Device, on page 960
- How to Configure an IPv6 Router Advertisement Guard Policy, on page 963
- **How to Configure an IPv6 DHCP Guard Policy** , on page 968
- How to Configure IPv6 Source Guard, on page 972
- How to Configure IPv6 Prefix Guard, on page 975
- Configuration Examples for IPv6 First Hop Security, on page 978
- Additional References, on page 979

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to <http://www.cisco.com/go/cfn>. An account on Cisco.com is not required.

Prerequisites for First Hop Security in IPv6

- You have configured the necessary IPv6 enabled SDM template.
- QoS should be enabled on the switch before configuring CoPP policies using **mls qos** command.

Restrictions for First Hop Security in IPv6

- The following restrictions apply when applying FHS policies to EtherChannel interfaces (Port Channels):
 - A physical port with an FHS policy attached cannot join an EtherChannel group.
 - An FHS policy cannot be attached to a physical port when it is a member of an EtherChannel group.
- By default, a snooping policy has a security-level of guard. When such a snooping policy is configured on an access switch, external IPv6 Router Advertisement (RA) or Dynamic Host Configuration Protocol for IPv6 (DHCPv6) server packets are blocked, even though the uplink port facing the router or DHCP server/relay is configured as a trusted port. To allow IPv6 RA or DHCPv6 server messages, do the following:
 - Apply an IPv6 RA-guard policy (for RA) or IPv6 DHCP-guard policy (for DHCP server messages) on the uplink port.
 - Configure a snooping policy with a lower security-level, for example glean or inspect. However, configuring a lower security level is not recommended with such a snooping policy, because benefits of First Hop security features are not effective.
- The following restrictions apply for CoPP policies with IPv6 SISF-based device tracking policies due to limitation reported in [CSCvk32439](#):
 - CoPP policies are required to limit IPv6 NDP traffic when IPv6 SISF policies are configured on the switch.
 - After NDP CoPP policies are configured, limited traffic hits CPU. To accommodate the total end points connected, the number of NDP CoPP policies should be slightly more than the number of users connected to each switch in a stack. If you configure NDP CoPP policies less than the number of end points connected to the switch, the IP allocation to the end point is delayed but is not ignored completely.



Note For example, if a stack of 5 switches has approximately 300 users, the NDP CoPP policies should be more than 300.

- The DHCPv6 (server-to-client and client-to-server) CoPP policies are required only if Lightweight DHCPv6 Relay Agent (LDRA) is configured under IPv6 SISF-based device tracking policies on the switch.

Information about First Hop Security in IPv6

First Hop Security in IPv6 (FHS IPv6) is a set of IPv6 security features, the policies of which can be attached to a physical interface, or a VLAN. An IPv6 software policy database service stores and accesses these policies. When a policy is configured or modified, the attributes of the policy are stored or updated in the software policy database, then applied as was specified. The following IPv6 policies are currently supported:

- IPv6 Snooping Policy—IPv6 Snooping Policy acts as a container policy that enables most of the features available with FHS in IPv6.
- IPv6 FHS Binding Table Content—A database table of IPv6 neighbors connected to the switch is created from information sources such as Neighbor Discovery (ND) protocol snooping. This database, or binding, table is used by various IPv6 guard features (such as IPv6 ND Inspection) to validate the link-layer address (LLA), the IPv4 or IPv6 address, and prefix binding of the neighbors to prevent spoofing and redirect attacks.
- IPv6 Neighbor Discovery Inspection—IPv6 ND inspection learns and secures bindings for stateless autoconfiguration addresses in Layer 2 neighbor tables. IPv6 ND inspection analyzes neighbor discovery messages in order to build a trusted binding table database and IPv6 neighbor discovery messages that do not conform are dropped. An ND message is considered trustworthy if its IPv6-to-Media Access Control (MAC) mapping is verifiable.

This feature mitigates some of the inherent vulnerabilities of the ND mechanism, such as attacks on DAD, address resolution, router discovery, and the neighbor cache.

- IPv6 Router Advertisement Guard—The IPv6 Router Advertisement (RA) guard feature enables the network administrator to block or reject unwanted or rogue RA guard messages that arrive at the network switch platform. RAs are used by routers to announce themselves on the link. The RA Guard feature analyzes the RAs and filters out bogus RAs sent by unauthorized routers. In host mode, all router advertisement and router redirect messages are disallowed on the port. The RA guard feature compares configuration information on the Layer 2 device with the information found in the received RA frame. Once the Layer 2 device has validated the content of the RA frame and router redirect frame against the configuration, it forwards the RA to its unicast or multicast destination. If the RA frame content is not validated, the RA is dropped.
- IPv6 DHCP Guard—The IPv6 DHCP Guard feature blocks reply and advertisement messages that come from unauthorized DHCPv6 servers and relay agents. IPv6 DHCP guard can prevent forged messages from being entered in the binding table and block DHCPv6 server messages when they are received on ports that are not explicitly configured as facing a DHCPv6 server or DHCP relay. To use this feature, configure a policy and attach it to an interface or a VLAN. To debug DHCP guard packets, use the **debug ipv6 snooping dhcp-guard** privileged EXEC command.
- IPv6 Source Guard—Like IPv4 Source Guard, IPv6 Source Guard validates the source address or prefix to prevent source address spoofing.

A source guard programs the hardware to allow or deny traffic based on source or destination addresses. It deals exclusively with data packet traffic.

The IPv6 source guard feature provides the ability to use the IPv6 binding table to install ACLs to prevent a host from sending packets with an invalid IPv6 source address.

To debug source-guard packets, use the `debug ipv6 snooping source-guard` privileged EXEC command.



Note The IPv6 ACL feature is supported only in the ingress direction; it is not supported in the egress direction.

The following restrictions apply:

- An FHS policy cannot be attached to a physical port when it is a member of an EtherChannel group.

- When IPv6 source guard is enabled on a switch port, NDP or DHCP snooping must be enabled on the interface to which the switch port belongs. Otherwise, all data traffic from this port will be blocked.
- An IPv6 source guard policy cannot be attached to a VLAN. It is supported only at the interface level.
- When you configure IPv4 and IPv6 source guard together on an interface, it is recommended to use **ip verify source mac-check** instead of **ip verify source**. IPv4 connectivity on a given port might break due to two different filtering rules set — one for IPv4 (IP-filter) and the other for IPv6 (IP-MAC filter).
- You cannot use IPv6 Source Guard and Prefix Guard together. When you attach the policy to an interface, it should be "validate address" or "validate prefix" but not both.
- PVLAN and Source/Prefix Guard cannot be applied together.

For more information on IPv6 Source Guard, see the [IPv6 Source Guard](#) chapter of the Cisco IOS IPv6 Configuration Guide Library on Cisco.com.

- IPv6 Prefix Guard—The IPv6 prefix guard feature works within the IPv6 source guard feature, to enable the device to deny traffic originated from non-topologically correct addresses. IPv6 prefix guard is often used when IPv6 prefixes are delegated to devices (for example, home gateways) using DHCP prefix delegation. The feature discovers ranges of addresses assigned to the link and blocks any traffic sourced with an address outside this range.

For more information on IPv6 Prefix Guard, see the [IPv6 Prefix Guard](#) chapter of the Cisco IOS IPv6 Configuration Guide Library on Cisco.com.

- IPv6 Destination Guard—The IPv6 destination guard feature works with IPv6 neighbor discovery to ensure that the device performs address resolution only for those addresses that are known to be active on the link. It relies on the address glean functionality to populate all destinations active on the link into the binding table and then blocks resolutions before they happen when the destination is not found in the binding table.



Note IPv6 Destination Guard is recommended only on Layer 3. It is not recommended on Layer 2.

For more information about IPv6 Destination Guard, see the [IPv6 Destination Guard](#) chapter of the Cisco IOS IPv6 Configuration Guide Library on Cisco.com.

- IPv6 Neighbor Discovery Multicast Suppress—The IPv6 Neighbor Discovery multicast suppress feature is an IPv6 snooping feature that runs on a switch or a wireless controller and is used to reduce the amount of control traffic necessary for proper link operations.
- DHCPv6 Relay—Lightweight DHCPv6 Relay Agent—The DHCPv6 Relay—Lightweight DHCPv6 Relay Agent feature allows relay agent information to be inserted by an access node that performs a link-layer bridging (non-routing) function. Lightweight DHCPv6 Relay Agent (LDRA) functionality can be implemented in existing access nodes, such as DSL access multiplexers (DSLAMs) and Ethernet switches, that do not support IPv6 control or routing functions. LDRA is used to insert relay-agent options in DHCP version 6 (DHCPv6) message exchanges primarily to identify client-facing interfaces. LDRA functionality can be enabled on an interface and on a VLAN.



Note If an LDRA device is directly connected to a client, the interface must have the pool configuration to fetch the specific subnet or link information at the server side. In this case, if the LDRA device is present in different subnets or links, the server may not be able to fetch the correct subnet. You can now configure the pool name in the interface so as to choose the proper subnet or link for the client.

For more information about DHCPv6 Relay, See the [DHCPv6 Relay—Lightweight DHCPv6 Relay Agent](#) section of the IP Addressing: DHCP Configuration Guide, Cisco IOS Release 15.1SG.

How to Configure an IPv6 Snooping Policy

Beginning in privileged EXEC mode, follow these steps to configure IPv6 Snooping Policy :

SUMMARY STEPS

1. **configure terminal**
2. **ipv6 snooping policy** *policy-name*
3. **{[default] | [device-role {node | switch}] | [limit address-count value] | [no] | [protocol {dhcp | ndp}] | [security-level {glean | guard | inspect}] | [tracking {disable [stale-lifetime [seconds | infinite]] | enable [reachable-lifetime [seconds | infinite]]}] | [trusted-port] }**
4. **end**
5. **show ipv6 snooping policy** *policy-name*

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters the global configuration mode.
Step 2	ipv6 snooping policy <i>policy-name</i> Example: Device(config)# ipv6 snooping policy example_policy	Creates a snooping policy and enters IPv6 Snooping Policy Configuration mode.
Step 3	{[default] [device-role {node switch}] [limit address-count value] [no] [protocol {dhcp ndp}] [security-level {glean guard inspect}] [tracking {disable [stale-lifetime [seconds infinite]] enable [reachable-lifetime [seconds infinite]]}] [trusted-port] } Example: Device(config-ipv6-snooping)# security-level inspect Example:	Enables data address glean, validates messages against various criteria, specifies the security level for messages. <ul style="list-style-type: none"> • (Optional) default—Sets all to default options. • (Optional) device-role {node} switch—Specifies the role of the device attached to the port. Default is node. • (Optional) limit address-count value—Limits the number of addresses allowed per target.

	Command or Action	Purpose
	Device (config-ipv6-snooping) # trusted-port	<ul style="list-style-type: none"> • (Optional) no—Negates a command or sets it to defaults. • (Optional) protocol {dhcp ndp}—Specifies which protocol should be redirected to the snooping feature for analysis. The default, is dhcp and ndp. To change the default, use the no protocol command. • (Optional) security-level {glean guard inspect}—Specifies the level of security enforced by the feature. Default is guard. <ul style="list-style-type: none"> glean—Gleans addresses from messages and populates the binding table without any verification. guard—Gleans addresses and inspects messages. In addition, it rejects RA and DHCP server messages. This is the default option. inspect—Gleans addresses, validates messages for consistency and conformance, and enforces address ownership. • (Optional) tracking {disable enable}—Overrides the default tracking behavior and specifies a tracking option. • (Optional) trusted-port—Sets up a trusted port. It disables the guard on applicable targets. Bindings learned through a trusted port have preference over bindings learned through any other port. A trusted port is given preference in case of a collision while making an entry in the table.
Step 4	end Example: Device (config-ipv6-snooping) # exit	Exits configuration modes to Privileged EXEC mode.
Step 5	show ipv6 snooping policy <i>policy-name</i> Example: Device# show ipv6 snooping policy example_policy	Displays the snooping policy configuration.

What to do next

Attach an IPv6 Snooping policy to interfaces or VLANs.

How to Attach an IPv6 Snooping Policy to an Interface

Beginning in privileged EXEC mode, follow these steps to attach an IPv6 Snooping policy on an interface or VLAN:

SUMMARY STEPS

1. **configure terminal**
2. **interface** Interface_type stack/module/port
3. **switchport**
4. **ipv6 snooping** [attach-policy policy_name [vlan {vlan_id} | add vlan_ids | exceptvlan_ids | none | remove vlan_ids] | vlan {vlan_id} | add vlan_ids | exceptvlan_ids | none | remove vlan_ids | all]
5. **do show running-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters the global configuration mode.
Step 2	interface Interface_type stack/module/port Example: Device(config)# interface gigabitethernet 1/1/4	Specifies an interface type and identifier; enters the interface configuration mode.
Step 3	switchport Example: Device(config-if)# switchport	Enters the Switchport mode. Note To configure Layer 2 parameters, if the interface is in Layer 3 mode, you must enter the switchport interface configuration command without any parameters to put the interface into Layer 2 mode. This shuts down the interface and then re-enables it, which might generate messages on the device to which the interface is connected. When you put an interface that is in Layer 3 mode into Layer 2 mode, the previous configuration information related to the affected interface might be lost, and the interface is returned to its default configuration. The command prompt displays as (config-if)# in Switchport configuration mode.
Step 4	ipv6 snooping [attach-policy policy_name [vlan {vlan_id} add vlan_ids exceptvlan_ids none remove vlan_ids] vlan {vlan_id} add vlan_ids exceptvlan_ids none remove vlan_ids all] Example:	Attaches a custom ipv6 snooping policy to the interface or the specified VLANs on the interface. To attach the default policy to the interface, use the ipv6 snooping command without the attach-policy keyword. To attach the default policy to VLANs on the interface, use the ipv6 snooping vlan command. The default policy is, security-level guard , device-role node , protocol ndp and dhcp .

	Command or Action	Purpose
	<pre>Device(config-if)# ipv6 snooping or Device(config-if)# ipv6 snooping attach-policy example_policy or Device(config-if)# ipv6 snooping vlan 111,112 or Device(config-if)# ipv6 snooping attach-policy example_policy vlan 111,112</pre>	
Step 5	<p>do show running-config</p> <p>Example:</p> <pre>Device#(config-if)# do show running-config</pre>	Verifies that the policy is attached to the specified interface without exiting the interface configuration mode.

How to Attach an IPv6 Snooping Policy to a Layer 2 EtherChannel Interface

Beginning in privileged EXEC mode, follow these steps to attach an IPv6 Snooping policy on an EtherChannel interface or VLAN:

Procedure

	Command or Action	Purpose
Step 1	<p>configure terminal</p> <p>Example:</p> <pre>Device# configure terminal</pre>	Enters the global configuration mode.
Step 2	<p>interface range <i>Interface_name</i></p> <p>Example:</p> <pre>Device(config)# interface range Po11</pre>	<p>Specify the port-channel interface name assigned when the EtherChannel was created. Enters the interface range configuration mode.</p> <p>Tip Enter the do show interfaces summary command for quick reference to interface names and types.</p>
Step 3	<p>ipv6 snooping [attach-policy <i>policy_name</i> [vlan {<i>vlan_ids</i> add <i>vlan_ids</i> except <i>vlan_ids</i> none remove <i>vlan_ids</i> all}] vlan [{<i>vlan_ids</i> add <i>vlan_ids</i> except <i>vlan_ids</i> none remove <i>vlan_ids</i> all}]]</p> <p>Example:</p> <pre>Device(config-if-range)# ipv6 snooping attach-policy example_policy or</pre>	Attaches the IPv6 Snooping policy to the interface or the specified VLANs on that interface. The default policy is attached if the attach-policy option is not used.

	Command or Action	Purpose
	<pre>Device(config-if-range)# ipv6 snooping attach-policy example_policy vlan 222,223,224 or Device(config-if-range)#ipv6 snooping vlan 222, 223,224</pre>	
Step 4	<p>do show running-config interfaceportchannel_interface_name</p> <p>Example:</p> <pre>Device#(config-if-range)# do show running-config int poll</pre>	Confirms that the policy is attached to the specified interface without exiting the configuration mode.

How to Configure the IPv6 Binding Table Content

Beginning in privileged EXEC mode, follow these steps to configure IPv6 Binding Table Content :

SUMMARY STEPS

1. **configure terminal**
2. **[no] ipv6 neighbor binding [vlan vlan-id {ipv6-address interface interface_type stack/module/port hw_address [reachable-lifetimevalue [seconds | default | infinite] | [tracking{ [default | disable] [reachable-lifetimevalue [seconds | default | infinite] | [enable [reachable-lifetimevalue [seconds | default | infinite] | [retry-interval {seconds| default [reachable-lifetimevalue [seconds | default | infinite] }]**
3. **[no] ipv6 neighbor binding max-entries number [mac-limit number | port-limit number [mac-limit number] | vlan-limit number [mac-limit number] | [port-limit number [mac-limit number]]]**
4. **ipv6 neighbor binding logging**
5. **exit**
6. **show ipv6 neighbor binding**

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p>configure terminal</p> <p>Example:</p> <pre>Device# configure terminal</pre>	Enters the global configuration mode.
Step 2	<pre>[no] ipv6 neighbor binding [vlan vlan-id {ipv6-address interface interface_type stack/module/port hw_address [reachable-lifetimevalue [seconds default infinite] [tracking{ [default disable] [reachable-lifetimevalue [seconds default infinite] [enable [reachable-lifetimevalue [seconds default infinite] </pre>	Adds a static entry to the binding table database.

	Command or Action	Purpose
	<pre>[retry-interval {seconds} default [reachable-lifetimevalue seconds default infinite]] Example: Device(config)# ipv6 neighbor binding</pre>	
Step 3	<pre>[no] ipv6 neighbor binding max-entries number [mac-limit number port-limit number [mac-limit number] vlan-limit number [[mac-limit number] [port-limit number [mac-limitnumber]]]] Example: Device(config)# ipv6 neighbor binding max-entries 30000</pre>	Specifies the maximum number of entries that are allowed to be inserted in the binding table cache.
Step 4	<pre>ipv6 neighbor binding logging Example: Device(config)# ipv6 neighbor binding logging</pre>	Enables the logging of binding table main events.
Step 5	<pre>exit Example: Device(config)# exit</pre>	Exits global configuration mode, and places the router in privileged EXEC mode.
Step 6	<pre>show ipv6 neighbor binding Example: Device# show ipv6 neighbor binding</pre>	Displays contents of a binding table.

How to Configure an IPv6 Neighbor Discovery Inspection Policy

Beginning in privileged EXEC mode, follow these steps to configure an IPv6 ND Inspection Policy:

SUMMARY STEPS

1. **configure terminal**
2. **[no]ipv6 nd inspection policy *policy-name***
3. **device-role {host | monitor | router | switch}**
4. **drop-unsecure**
5. **limit address-count *value***
6. **sec-level minimum *value***
7. **tracking {enable [reachable-lifetime {*value* | infinite}] | disable [stale-lifetime {*value* | infinite}]}**
8. **trusted-port**
9. **validate source-mac**
10. **no {device-role | drop-unsecure | limit address-count | sec-level minimum | tracking | trusted-port | validate source-mac}**
11. **default {device-role | drop-unsecure | limit address-count | sec-level minimum | tracking | trusted-port | validate source-mac}**

12. do show ipv6 nd inspection policy *policy_name*

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters the global configuration mode.
Step 2	[no]ipv6 nd inspection policy <i>policy-name</i> Example: Device(config)# ipv6 nd inspection policy example_policy	Specifies the ND inspection policy name and enters ND Inspection Policy configuration mode.
Step 3	device-role {host monitor router switch} Example: Device(config-nd-inspection)# device-role switch	Specifies the role of the device attached to the port. The default is host .
Step 4	drop-unsecure Example: Device(config-nd-inspection)# drop-unsecure	Drops messages with no or invalid options or an invalid signature.
Step 5	limit address-count <i>value</i> Example: Device(config-nd-inspection)# limit address-count 1000	Enter 1–10,000.
Step 6	sec-level minimum <i>value</i> Example: Device(config-nd-inspection)# limit address-count 1000	Specifies the minimum security level parameter value when Cryptographically Generated Address (CGA) options are used.
Step 7	tracking {enable [reachable-lifetime {<i>value</i> infinite}] disable [stale-lifetime {<i>value</i> infinite}]} Example: Device(config-nd-inspection)# tracking disable stale-lifetime infinite	Overrides the default tracking policy on a port.
Step 8	trusted-port Example: Device(config-nd-inspection)# trusted-port	Configures a port to become a trusted port.
Step 9	validate source-mac Example: Device(config-nd-inspection)# validate source-mac	Checks the source media access control (MAC) address against the link-layer address.

	Command or Action	Purpose
Step 10	no {device-role drop-unsecure limit address-count sec-level minimum tracking trusted-port validate source-mac} Example: Device(config-nd-inspection)# no validate source-mac	Remove the current configuration of a parameter with the no form of the command.
Step 11	default {device-role drop-unsecure limit address-count sec-level minimum tracking trusted-port validate source-mac} Example: Device(config-nd-inspection)# default limit address-count	Restores configuration to the default values.
Step 12	do show ipv6 nd inspection policy <i>policy_name</i> Example: Device(config-nd-inspection)# do show ipv6 nd inspection policy example_policy	Verifies the ND Inspection Configuration without exiting ND inspection configuration mode.

How to Attach an IPv6 Neighbor Discovery Inspection Policy to an Interface

Beginning in privileged EXEC mode, follow these steps to attach an IPv6 ND Inspection policy to an interface or VLANs on an interface :

SUMMARY STEPS

1. **configure terminal**
2. **interface** Interface_type *stack/module/port*
3. **ipv6 nd inspection** [**attach-policy** *policy_name* [**vlan** {*vlan_ids* | **add** *vlan_ids* | **except** *vlan_ids* | **none** | **remove** *vlan_ids* | **all** }] | **vlan** [{*vlan_ids* | **add** *vlan_ids* | **except** *vlan_ids* | **none** | **remove** *vlan_ids* | **all** }]]
4. **do show running-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters the global configuration mode.
Step 2	interface Interface_type <i>stack/module/port</i> Example: Device(config)# interface gigabitethernet 1/1/4	Specifies an interface type and identifier; enters the interface configuration mode.

	Command or Action	Purpose
Step 3	<p>ipv6 nd inspection [attach-policy <i>policy_name</i> [vlan {<i>vlan_ids</i> add <i>vlan_ids</i> except <i>vlan_ids</i> none remove <i>vlan_ids</i> all}] vlan [{<i>vlan_ids</i> add <i>vlan_ids</i> except <i>vlan_ids</i> none remove <i>vlan_ids</i> all}]]</p> <p>Example:</p> <pre>Device(config-if)# ipv6 nd inspection attach-policy example_policy</pre> <p>or</p> <pre>Device(config-if)# ipv6 nd inspection attach-policy example_policy vlan 222,223,224</pre> <p>or</p> <pre>Device(config-if)# ipv6 nd inspection vlan 222, 223,224</pre>	Attaches the Neighbor Discovery Inspection policy to the interface or the specified VLANs on that interface. The default policy is attached if the attach-policy option is not used.
Step 4	<p>do show running-config</p> <p>Example:</p> <pre>Device#(config-if)# do show running-config</pre>	Verifies that the policy is attached to the specified interface without exiting the interface configuration mode.

How to Attach an IPv6 Neighbor Discovery Inspection Policy to a Layer 2 EtherChannel Interface

Beginning in privileged EXEC mode, follow these steps to attach an IPv6 Neighbor Discovery Inspection policy on an EtherChannel interface or VLAN:

SUMMARY STEPS

1. **configure terminal**
2. **interface range** *Interface_name*
3. **ipv6 nd inspection** [**attach-policy** *policy_name* [**vlan** {*vlan_ids* | **add** *vlan_ids* | **except** *vlan_ids* | **none** | **remove** *vlan_ids* | **all**}] | **vlan** [{*vlan_ids* | **add** *vlan_ids* | **except** *vlan_ids* | **none** | **remove** *vlan_ids* | **all**}]]
4. **do show running-config interfaceportchannel_interface_name**

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p>configure terminal</p> <p>Example:</p> <pre>Device# configure terminal</pre>	Enters the global configuration mode.

	Command or Action	Purpose
Step 2	interface range <i>Interface_name</i> Example: Device(config)# interface Po11	Specify the port-channel interface name assigned when the EtherChannel was created. Enters the interface range configuration mode. Tip Enter the do show interfaces summary command for quick reference to interface names and types.
Step 3	ipv6 nd inspection [attach-policy <i>policy_name</i> [vlan { <i>vlan_ids</i> add <i>vlan_ids</i> except <i>vlan_ids</i> none remove <i>vlan_ids</i> all }] vlan [{ <i>vlan_ids</i> add <i>vlan_ids</i> except <i>vlan_ids</i> none remove <i>vlan_ids</i> all }]] Example: Device(config-if-range)# ipv6 nd inspection attach-policy example_policy or Device(config-if-range)# ipv6 nd inspection attach-policy example_policy vlan 222,223,224 or Device(config-if-range)# ipv6 nd inspection vlan 222, 223,224	Attaches the ND Inspection policy to the interface or the specified VLANs on that interface. The default policy is attached if the attach-policy option is not used.
Step 4	do show running-config interfaceportchannel_interface_name Example: Device#(config-if-range)# do show running-config int po11	Confirms that the policy is attached to the specified interface without exiting the configuration mode.

How to Attach an IPv6 Neighbor Discovery Multicast Suppress Policy on a Device

To attach an IPV6 Neighbor Discovery Multicast Suppress policy on a device, complete the following steps:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ipv6 nd suppress policy** *policy-name*
4. **mode dad-proxy**
5. **mode full-proxy**
6. **mode mc-proxy**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal	Enters the global configuration mode.
Step 3	ipv6 nd suppress policy <i>policy-name</i>	Defines the Neighbor Discovery suppress policy name and enters Neighbor Discovery suppress policy configuration mode.
Step 4	mode dad-proxy	Enables Neighbor Discovery suppress in IPv6 DAD proxy mode.
Step 5	mode full-proxy	Enables Neighbor Discovery suppress to proxy multicast and unicast Neighbor Solicitation messages.
Step 6	mode mc-proxy	Enables Neighbor Discovery suppress to proxy multicast Neighbor Solicitation messages.

How to Attach an IPv6 Neighbor Discovery Multicast Suppress Policy on an Interface

To attach an IPv6 Neighbor Discovery Multicast Suppress policy on an interface, complete the following steps:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. Perform one of the following tasks:
 - **interface** *type number*
 - **ipv6 nd inspection** [**attach-policy** *policy_name* [**vlan** { **add** | **except** | **none** | **remove** | **all** } *vlan* [*vlan1, vlan2, vlan3...*]]]

OR

 - **vlan configuration** *vlan-id*
 - **ipv6 nd inspection** [**attach-policy** *policy_name* [**vlan** { **add** | **except** | **none** | **remove** | **all** } *vlan* [*vlan1, vlan2, vlan3...*]]]
4. **exit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal	Enters the global configuration mode.
Step 3	Perform one of the following tasks: <ul style="list-style-type: none"> • interface <i>type number</i> • ipv6 nd inspection [attach-policy <i>policy_name</i> [vlan { add except none remove all } <i>vlan</i> [<i>vlan1</i>, <i>vlan2</i>, <i>vlan3</i>...]]] OR <ul style="list-style-type: none"> • vlan configuration <i>vlan-id</i> • ipv6 nd inspection [attach-policy <i>policy_name</i> [vlan { add except none remove all } <i>vlan</i> [<i>vlan1</i>, <i>vlan2</i>, <i>vlan3</i>...]]] 	Specifies an interface type and number, and places the device in interface configuration mode. Attaches the IPv6 Neighbor Discovery Multicast Policy to an interface or a VLAN.
Step 4	exit	Exits the interface configuration mode.

How to Attach an IPv6 Neighbor Discovery Multicast Suppress Policy to a Layer 2 EtherChannel Interface

To attach an IPv6 Neighbor Discovery Multicast Suppress policy on an EtherChannel interface, complete the following steps:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. Perform one of the following tasks:
 - **interface port-channel** *port-channel-number*
 - **ipv6 nd inspection** [**attach-policy** *policy_name* [**vlan** { **add** | **except** | **none** | **remove** | **all** } *vlan* [*vlan1*, *vlan2*, *vlan3*...]]]
 OR
 - **vlan configuration** *vlan-id*
 - **ipv6 nd inspection** [**attach-policy** *policy_name* [**vlan** { **add** | **except** | **none** | **remove** | **all** } *vlan* [*vlan1*, *vlan2*, *vlan3*...]]]
4. **exit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal	Enters the global configuration mode.
Step 3	Perform one of the following tasks: <ul style="list-style-type: none"> • interface port-channel <i>port-channel-number</i> • ipv6 nd inspection [attach-policy <i>policy_name</i> [vlan { add except none remove all } <i>vlan</i> [<i>vlan1</i>, <i>vlan2</i>, <i>vlan3</i>...]]] OR <ul style="list-style-type: none"> • vlan configuration <i>vlan-id</i> • ipv6 nd inspection [attach-policy <i>policy_name</i> [vlan { add except none remove all } <i>vlan</i> [<i>vlan1</i>, <i>vlan2</i>, <i>vlan3</i>...]]] 	Specifies an interface type and port number and places the switch in the port channel configuration mode. Attaches the IPv6 Neighbor Discovery Multicast Policy to an interface or a VLAN.
Step 4	exit	Exits the interface configuration mode.

How to Configure an IPv6 Router Advertisement Guard Policy

Beginning in privileged EXEC mode, follow these steps to configure an IPv6 Router Advertisement policy :

SUMMARY STEPS

1. **configure terminal**
2. **[no]ipv6 nd rguard policy** *policy-name*
3. **[no]device-role** { **host** | **monitor** | **router** | **switch** }
4. **[no]hop-limit** { **maximum** | **minimum** } *value*
5. **[no]managed-config-flag** { **off** | **on** }
6. **[no]match** { **ipv6 access-list** *list* | **ra prefix-list** *list* }
7. **[no]other-config-flag** { **on** | **off** }
8. **[no]router-preference maximum** { **high** | **medium** | **low** }
9. **[no]trusted-port**
10. **default** { **device-role** | **hop-limit** { **maximum** | **minimum** } | **managed-config-flag** | **match** { **ipv6 access-list** | **ra prefix-list** } | **other-config-flag** | **router-preference maximum** | **trusted-port** }
11. **do show ipv6 nd rguard policy** *policy_name*

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters the global configuration mode.
Step 2	[no]ipv6 nd raguard policy <i>policy-name</i> Example: Device(config)# ipv6 nd raguard policy example_policy	Specifies the RA Guard policy name and enters RA Guard Policy configuration mode.
Step 3	[no]device-role {host monitor router switch} Example: Device(config-nd-raguard) # device-role switch	Specifies the role of the device attached to the port. The default is host .
Step 4	[no]hop-limit {maximum minimum} <i>value</i> Example: Device(config-nd-raguard) # hop-limit maximum 33	(1–255) Range for Maximum and Minimum Hop Limit values. Enables filtering of Router Advertisement messages by the Hop Limit value. A rogue RA message may have a low Hop Limit value (equivalent to the IPv4 Time to Live) that when accepted by the host, prevents the host from generating traffic to destinations beyond the rogue RA message generator. An RA message with an unspecified Hop Limit value is blocked. If not configured, this filter is disabled. Configure minimum to block RA messages with Hop Limit values lower than the value you specify. Configure maximum to block RA messages with Hop Limit values greater than the value you specify.
Step 5	[no]managed-config-flag {off on} Example: Device(config-nd-raguard) # managed-config-flag on	Enables filtering of Router Advertisement messages by the Managed Address Configuration, or "M" flag field. A rogue RA message with an M field of 1 can cause a host to use a rogue DHCPv6 server. If not configured, this filter is disabled. On —Accepts and forwards RA messages with an M value of 1, blocks those with 0. Off —Accepts and forwards RA messages with an M value of 0, blocks those with 1.
Step 6	[no]match {ipv6 access-list <i>list</i> ra prefix-list <i>list</i>} Example: Device(config-nd-raguard) # match ipv6 access-list example_list	Matches a specified prefix list or access list.

	Command or Action	Purpose
Step 7	<p><code>[no]other-config-flag {on off}</code></p> <p>Example:</p> <pre>Device(config-nd-raguard)# other-config-flag on</pre>	<p>Enables filtering of Router Advertisement messages by the Other Configuration, or "O" flag field. A rogue RA message with an O field of 1 can cause a host to use a rogue DHCPv6 server. If not configured, this filter is disabled.</p> <p>On—Accepts and forwards RA messages with an O value of 1, blocks those with 0.</p> <p>Off—Accepts and forwards RA messages with an O value of 0, blocks those with 1.</p>
Step 8	<p><code>[no]router-preference maximum {high medium low}</code></p> <p>Example:</p> <pre>Device(config-nd-raguard)# router-preference maximum high</pre>	<p>Enables filtering of Router Advertisement messages by the Router Preference flag. If not configured, this filter is disabled.</p> <ul style="list-style-type: none"> • high—Accepts RA messages with the Router Preference set to high, medium, or low. • medium—Blocks RA messages with the Router Preference set to high. • low—Blocks RA messages with the Router Preference set to medium and high.
Step 9	<p><code>[no]trusted-port</code></p> <p>Example:</p> <pre>Device(config-nd-raguard)# trusted-port</pre>	<p>When configured as a trusted port, all attached devices are trusted, and no further message verification is performed.</p>
Step 10	<p><code>default {device-role hop-limit {maximum minimum} managed-config-flag match {ipv6 access-list ra prefix-list } other-config-flag router-preference maximum trusted-port}</code></p> <p>Example:</p> <pre>Device(config-nd-raguard)# default hop-limit</pre>	<p>Restores a command to its default value.</p>
Step 11	<p><code>do show ipv6 nd raguard policy <i>policy_name</i></code></p> <p>Example:</p> <pre>Device(config-nd-raguard)# do show ipv6 nd raguard policy example_policy</pre>	<p>(Optional)—Displays the ND Guard Policy configuration without exiting the RA Guard policy configuration mode.</p>

How to Attach an IPv6 Router Advertisement Guard Policy to an Interface

Beginning in privileged EXEC mode, follow these steps to attach an IPv6 Router Advertisement policy to an interface or to VLANs on the interface :

SUMMARY STEPS

1. configure terminal

2. **interface** Interface_type stack/module/port
3. **ipv6 nd rguard** [attach-policy policy_name [vlan {vlan_ids | add vlan_ids | except vlan_ids | none | remove vlan_ids | all}] | vlan [{vlan_ids | add vlan_ids | exceptvlan_ids | none | remove vlan_ids | all}]
4. **do show running-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters the global configuration mode.
Step 2	interface Interface_type stack/module/port Example: Device(config)# interface gigabitethernet 1/1/4	Specifies an interface type and identifier; enters the interface configuration mode.
Step 3	ipv6 nd rguard [attach-policy policy_name [vlan {vlan_ids add vlan_ids except vlan_ids none remove vlan_ids all}] vlan [{vlan_ids add vlan_ids exceptvlan_ids none remove vlan_ids all}] Example: Device(config-if)# ipv6 nd rguard attach-policy example_policy or Device(config-if)# ipv6 nd rguard attach-policy example_policy vlan 222,223,224 or Device(config-if)# ipv6 nd rguard vlan 222,223,224	Attaches the Neighbor Discovery Inspection policy to the interface or the specified VLANs on that interface. The default policy is attached if the attach-policy option is not used.
Step 4	do show running-config Example: Device#(config-if)# do show running-config	Confirms that the policy is attached to the specified interface without exiting the configuration mode.

How to Attach an IPv6 Router Advertisement Guard Policy to a Layer 2 EtherChannel Interface

Beginning in privileged EXEC mode, follow these steps to attach an IPv6 Router Advertisement Guard Policy on an EtherChannel interface or VLAN:

SUMMARY STEPS

1. **configure terminal**
2. **interface range** *Interface_name*
3. **ipv6 nd raguard** [**attach-policy** *policy_name* [**vlan** {*vlan_ids* | **add** *vlan_ids* | **except** *vlan_ids* | **none** | **remove** *vlan_ids* | **all**}] | **vlan** [{*vlan_ids* | **add** *vlan_ids* | **except** *vlan_ids* | **none** | **remove** *vlan_ids* | **all**}]]
4. **do show running-config interface***portchannel_interface_name*

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters the global configuration mode.
Step 2	interface range <i>Interface_name</i> Example: Device(config)# interface Po11	Specify the port-channel interface name assigned when the EtherChannel was created. Enters the interface range configuration mode. Tip Enter the do show interfaces summary command for quick reference to interface names and types.
Step 3	ipv6 nd raguard [attach-policy <i>policy_name</i> [vlan { <i>vlan_ids</i> add <i>vlan_ids</i> except <i>vlan_ids</i> none remove <i>vlan_ids</i> all }] vlan [{ <i>vlan_ids</i> add <i>vlan_ids</i> except <i>vlan_ids</i> none remove <i>vlan_ids</i> all }]] Example: Device(config-if-range)# ipv6 nd raguard attach-policy example_policy or Device(config-if-range)# ipv6 nd raguard attach-policy example_policy vlan 222,223,224 or Device(config-if-range)# ipv6 nd raguard vlan 222,223,224	Attaches the RA Guard policy to the interface or the specified VLANs on that interface. The default policy is attached if the attach-policy option is not used.
Step 4	do show running-config interface <i>portchannel_interface_name</i> Example: Device#(config-if-range)# do show running-config int po11	Confirms that the policy is attached to the specified interface without exiting the configuration mode.

How to Configure an IPv6 DHCP Guard Policy

Beginning in privileged EXEC mode, follow these steps to configure an IPv6 DHCP (DHCPv6) Guard policy:

SUMMARY STEPS

1. **configure terminal**
2. **[no]ipv6 dhcp guard policy *policy-name***
3. **[no]device-role {client | server}**
4. **[no] match server access-list *ipv6-access-list-name***
5. **[no] match reply prefix-list *ipv6-prefix-list-name***
6. **[no]preference { max *limit* | min *limit* }**
7. **[no] trusted-port**
8. **default {device-role | trusted-port}**
9. **do show ipv6 dhcp guard policy *policy_name***

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters the global configuration mode.
Step 2	[no]ipv6 dhcp guard policy <i>policy-name</i> Example: Device(config)# ipv6 dhcp guard policy example_policy	Specifies the DHCPv6 Guard policy name and enters DHCPv6 Guard Policy configuration mode.
Step 3	[no]device-role {client server} Example: Device(config-dhcp-guard)# device-role server	(Optional) Filters out DHCPv6 replies and DHCPv6 advertisements on the port that are not from a device of the specified role. Default is client . <ul style="list-style-type: none"> • client—Default value, specifies that the attached device is a client. Server messages are dropped on this port. • server—Specifies that the attached device is a DHCPv6 server. Server messages are allowed on this port.
Step 4	[no] match server access-list <i>ipv6-access-list-name</i> Example: ;;Assume a preconfigured IPv6 Access List as follows: Device(config)# ipv6 access-list my_acls Device(config-ipv6-acl)# permit host FE80::A8BB:CCFF:FE01:F700 any	(Optional). Enables verification that the advertised DHCPv6 server or relay address is from an authorized server access list (The destination address in the access list is 'any'). If not configured, this check will be bypassed. An empty access list is treated as a permit all.

	Command or Action	Purpose
	<pre>;;configure DHCPv6 Guard to match approved access list. Device(config-dhcp-guard)# match server access-list my_acls</pre>	
Step 5	<p>[no] match reply prefix-list <i>ipv6-prefix-list-name</i></p> <p>Example:</p> <pre>;;Assume a preconfigured IPv6 prefix list as follows: Device(config)# ipv6 prefix-list my_prefix permit 2001:0DB8::/64 le 128 ;; Configure DHCPv6 Guard to match prefix Device(config-dhcp-guard)# match reply prefix-list my_prefix</pre>	(Optional) Enables verification of the advertised prefixes in DHCPv6 reply messages from the configured authorized prefix list. If not configured, this check will be bypassed. An empty prefix list is treated as a permit.
Step 6	<p>[no]preference{ <i>max limit</i> <i>min limit</i> }</p> <p>Example:</p> <pre>Device(config-dhcp-guard)# preference max 250 Device(config-dhcp-guard)#preference min 150</pre>	<p>Configure max and min when device-role is server to filter DHCPv6 server advertisements by the server preference value. The defaults permit all advertisements.</p> <p>max limit—(0 to 255) (Optional) Enables verification that the advertised preference (in preference option) is less than the specified limit. Default is 255. If not specified, this check will be bypassed.</p> <p>min limit—(0 to 255) (Optional) Enables verification that the advertised preference (in preference option) is greater than the specified limit. Default is 0. If not specified, this check will be bypassed.</p>
Step 7	<p>[no] trusted-port</p> <p>Example:</p> <pre>Device(config-dhcp-guard)# trusted-port</pre>	<p>(Optional) trusted-port—Sets the port to a trusted mode. No further policing takes place on the port.</p> <p>Note If you configure a trusted port then the device-role option is not available.</p>
Step 8	<p>default {<i>device-role</i> <i>trusted-port</i>}</p> <p>Example:</p> <pre>Device(config-dhcp-guard)# default device-role</pre>	(Optional) default —Sets a command to its defaults.
Step 9	<p>do show ipv6 dhcp guard policy <i>policy_name</i></p> <p>Example:</p> <pre>Device(config-dhcp-guard)# do show ipv6 dhcp guard policy example_policy</pre>	(Optional) Displays the configuration of the IPv6 DHCP guard policy without leaving the configuration submenu. Omitting the <i>policy_name</i> variable displays all DHCPv6 policies.

Example of DHCPv6 Guard Configuration

```
enable
configure terminal
ipv6 access-list acl1
```

```

permit host FE80::A8BB:CCFF:FE01:F700 any
ipv6 prefix-list abc permit 2001:0DB8::/64 le 128
ipv6 dhcp guard policy poll
device-role server
match server access-list acl1
match reply prefix-list abc
preference min 0
preference max 255
trusted-port
interface GigabitEthernet 0/2/0
switchport
ipv6 dhcp guard attach-policy poll vlan add 1
vlan 1
  ipv6 dhcp guard attach-policy poll
show ipv6 dhcp guard policy poll

```

How to Attach an IPv6 DHCP Guard Policy to an Interface or a VLAN on an Interface

Beginning in privileged EXEC mode, follow these steps to configure IPv6 Binding Table Content :

SUMMARY STEPS

1. **configure terminal**
2. **interface** Interface_type stack/module/port
3. **ipv6 dhcp guard** [**attach-policy** policy_name [**vlan** {vlan_ids | **add** vlan_ids | **except** vlan_ids | **none** | **remove** vlan_ids | **all** }] | **vlan** [{vlan_ids | **add** vlan_ids | **except** vlan_ids | **none** | **remove** vlan_ids | **all** }]
4. **do show running-config interface** Interface_type stack/module/port

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters the global configuration mode.
Step 2	interface Interface_type stack/module/port Example: Device(config)# interface gigabitethernet 1/1/4	Specifies an interface type and identifier; enters the interface configuration mode.
Step 3	ipv6 dhcp guard [attach-policy policy_name [vlan {vlan_ids add vlan_ids except vlan_ids none remove vlan_ids all }] vlan [{vlan_ids add vlan_ids except vlan_ids none remove vlan_ids all }] Example: Device(config-if)# ipv6 dhcp guard attach-policy example_policy or	Attaches the DHCP Guard policy to the interface or the specified VLANs on that interface. The default policy is attached if the attach-policy option is not used.

	Command or Action	Purpose
	<pre>Device(config-if)# ipv6 dhcp guard attach-policy example_policy vlan 222,223,224 or Device(config-if)# ipv6 dhcp guard vlan 222, 223,224</pre>	
Step 4	<p>do show running-config interface <i>Interface_type</i> <i>stack/module/port</i></p> <p>Example:</p> <pre>Device#(config-if)# do show running-config gig 1/1/4</pre>	Confirms that the policy is attached to the specified interface without exiting the configuration mode.

How to Attach an IPv6 DHCP Guard Policy to a Layer 2 EtherChannel Interface

Beginning in privileged EXEC mode, follow these steps to attach an IPv6 DHCP Guard policy on an EtherChannel interface or VLAN:

SUMMARY STEPS

1. **configure terminal**
2. **interface range** *Interface_name*
3. **ipv6 dhcp guard** [**attach-policy** *policy_name* [**vlan** {*vlan_ids* | **add** *vlan_ids* | **except** *vlan_ids* | **none** | **remove** *vlan_ids* | **all**}] | **vlan** [{*vlan_ids* | **add** *vlan_ids* | **except** *vlan_ids* | **none** | **remove** *vlan_ids* | **all**}]]
4. **do show running-config interface***portchannel_interface_name*

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p>configure terminal</p> <p>Example:</p> <pre>Device# configure terminal</pre>	Enters the global configuration mode.
Step 2	<p>interface range <i>Interface_name</i></p> <p>Example:</p> <pre>Device(config)# interface Po11</pre>	Specify the port-channel interface name assigned when the EtherChannel was created. Enters the interface range configuration mode. Tip Enter the do show interfaces summary command for quick reference to interface names and types.

	Command or Action	Purpose
Step 3	<p>ipv6 dhcp guard [attach-policy <i>policy_name</i> [vlan {<i>vlan_ids</i> add <i>vlan_ids</i> except <i>vlan_ids</i> none remove <i>vlan_ids</i> all}]] vlan [{<i>vlan_ids</i> add <i>vlan_ids</i> except <i>vlan_ids</i> none remove <i>vlan_ids</i> all}]</p> <p>Example:</p> <pre>Device(config-if-range)# ipv6 dhcp guard attach-policy example_policy or Device(config-if-range)# ipv6 dhcp guard attach-policy example_policy vlan 222,223,224 or Device(config-if-range)#ipv6 dhcp guard vlan 222, 223,224</pre>	Attaches the DHCP Guard policy to the interface or the specified VLANs on that interface. The default policy is attached if the attach-policy option is not used.
Step 4	<p>do show running-config interface<i>portchannel_interface_name</i></p> <p>Example:</p> <pre>Device#(config-if-range)# do show running-config int po11</pre>	Confirms that the policy is attached to the specified interface without exiting the configuration mode.

How to Configure IPv6 Source Guard

SUMMARY STEPS

1. enable
2. configure terminal
3. [no] ipv6 source-guard policy *policy_name*
4. [deny global-autoconf] [permit link-local] [default{. . .}] [exit] [no{. . .}]
5. end
6. show ipv6 source-guard policy *policy_name*

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p>enable</p> <p>Example:</p> <pre>Device> enable</pre>	<p>Enables privileged EXEC mode.</p> <ul style="list-style-type: none"> • Enter your password if prompted.

	Command or Action	Purpose
Step 2	configure terminal Example: Device# <code>configure terminal</code>	Enters the global configuration mode.
Step 3	[no] ipv6 source-guard policy <i>policy_name</i> Example: Device(config)# <code>ipv6 source-guard policy example_policy</code>	Specifies the IPv6 Source Guard policy name and enters IPv6 Source Guard policy configuration mode.
Step 4	[deny global-autoconf] [permit link-local] [default{...}] [exit] [no{...}] Example: Device(config-sisf-sourceguard)# <code>deny global-autoconf</code>	(Optional) Defines the IPv6 Source Guard policy. <ul style="list-style-type: none"> • deny global-autoconf—Denies data traffic from auto-configured global addresses. This is useful when all global addresses on a link are DHCP-assigned and the administrator wants to block hosts with self-configured addresses to send traffic. • permit link-local—Allows all data traffic that is sourced by a link-local address. <p>Note Trusted option under source guard policy is not supported.</p>
Step 5	end Example: Device(config-sisf-sourceguard)# <code>end</code>	Exits out of IPv6 Source Guard policy configuration mode.
Step 6	show ipv6 source-guard policy <i>policy_name</i> Example: Device# <code>show ipv6 source-guard policy example_policy</code>	Shows the policy configuration and all the interfaces where the policy is applied.

What to do next

Apply the IPv6 Source Guard policy to an interface.

How to Attach an IPv6 Source Guard Policy to an Interface

SUMMARY STEPS

1. `enable`
2. `configure terminal`
3. `interface` Interface_type *stack/module/port*
4. `ipv6 source-guard [attach-policy <policy_name>]`
5. `show ipv6 source-guard policy policy_name`

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters the global configuration mode.
Step 3	interface Interface_type <i>stack/module/port</i> Example: Device(config)# interface gigabitethernet 1/1/4	Specifies an interface type and identifier; enters the interface configuration mode.
Step 4	ipv6 source-guard [attach-policy <policy_name>] Example: Device(config-if)# ipv6 source-guard attach-policy example_policy	Attaches the IPv6 Source Guard policy to the interface. The default policy is attached if the attach-policy option is not used.
Step 5	show ipv6 source-guard policy policy_name Example: Device#(config-if)# show ipv6 source-guard policy example_policy	Shows the policy configuration and all the interfaces where the policy is applied.

How to attach an IPv6 Source Guard Policy to a Layer 2 EtherChannel Interface

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface port-channel** *port-channel-number*
4. **ipv6 source-guard [attach-policy <policy_name>]**
5. **show ipv6 source-guard policy** *policy_name*

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. • Enter your password if prompted.

	Command or Action	Purpose
Step 2	configure terminal Example: Device# configure terminal	Enters the global configuration mode.
Step 3	interface port-channel <i>port-channel-number</i> Example: Device (config)# interface Po4	Specifies an interface type and port number and places the switch in the port channel configuration mode.
Step 4	ipv6 source-guard [attach-policy <i><policy_name></i>] Example: Device(config-if) # ipv6 source-guard attach-policy example_policy	Attaches the IPv6 Source Guard policy to the interface. The default policy is attached if the attach-policy option is not used.
Step 5	show ipv6 source-guard policy <i>policy_name</i> Example: Device(config-if) # show ipv6 source-guard policy example_policy	Shows the policy configuration and all the interfaces where the policy is applied.

How to Configure IPv6 Prefix Guard



Note To allow routing protocol control packets sourced by a link-local address when prefix guard is applied, enable the permit link-local command in the source-guard policy configuration mode.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. [**no**] **ipv6 source-guard policy** *source-guard-policy*
4. [**no**] **validate address**
5. **validate prefix**
6. **exit**
7. **show ipv6 source-guard policy** [*source-guard-policy*]

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.

	Command or Action	Purpose
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	[no] ipv6 source-guard policy source-guard-policy Example: Device(config)# ipv6 source-guard policy my_snooping_policy	Defines an IPv6 source-guard policy name and enters switch integrated security features source-guard policy configuration mode.
Step 4	[no] validate address Example: Device(config-sisf-sourceguard)# no validate address	Disables the validate address feature and enables the IPv6 prefix guard feature to be configured.
Step 5	validate prefix Example: Device(config-sisf-sourceguard)# validate prefix	Enables IPv6 source guard to perform the IPv6 prefix-guard operation.
Step 6	exit Example: Device(config-sisf-sourceguard)# exit	Exits switch integrated security features source-guard policy configuration mode and returns to privileged EXEC mode.
Step 7	show ipv6 source-guard policy [source-guard-policy] Example: Device# show ipv6 source-guard policy policy1	Displays the IPv6 source-guard policy configuration.

How to Attach an IPv6 Prefix Guard Policy to an Interface

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** Interface_type *stack/module/port*
4. **ipv6 source-guard attach-policy** policy_name
5. **show ipv6 source-guard policy** policy_name

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example:	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.

	Command or Action	Purpose
	Device> <code>enable</code>	
Step 2	configure terminal Example: Device# <code>configure terminal</code>	Enters the global configuration mode.
Step 3	interface <i>Interface_type stack/module/port</i> Example: Device(config)# <code>interface gigabitethernet 1/1/4</code>	Specifies an interface type and identifier; enters the interface configuration mode.
Step 4	ipv6 source-guard attach-policy <i>policy_name</i> Example: Device(config-if)# <code>ipv6 source-guard attach-policy example_policy</code>	Attaches the IPv6 Source Guard policy to the interface. The default policy is attached if the attach-policy option is not used.
Step 5	show ipv6 source-guard policy <i>policy_name</i> Example: Device(config-if)# <code>show ipv6 source-guard policy example_policy</code>	Shows the policy configuration and all the interfaces where the policy is applied.

How to attach an IPv6 Prefix Guard Policy to a Layer 2 EtherChannel Interface

SUMMARY STEPS

1. `enable`
2. `configure terminal`
3. `interface port-channel port-channel-number`
4. `ipv6 source-guard [attach-policy <policy_name>]`
5. `show ipv6 source-guard policy policy_name`

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> <code>enable</code>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# <code>configure terminal</code>	Enters the global configuration mode.

	Command or Action	Purpose
Step 3	interface port-channel <i>port-channel-number</i> Example: Device (config)# interface Po4	Specifies an interface type and port number and places the switch in the port channel configuration mode.
Step 4	ipv6 source-guard [attach-policy < <i>policy_name</i> >] Example: Device(config-if)# ipv6 source-guard attach-policy example_policy	Attaches the IPv6 Source Guard policy to the interface. The default policy is attached if the attach-policy option is not used.
Step 5	show ipv6 source-guard policy <i>policy_name</i> Example: Device(config-if)# show ipv6 source-guard policy example_policy	Shows the policy configuration and all the interfaces where the policy is applied.

Configuration Examples for IPv6 First Hop Security

Examples: How to attach an IPv6 Source Guard Policy to a Layer 2 EtherChannel Interface

The following example shows how to attach an IPv6 Source Guard Policy to a Layer 2 EtherChannel Interface:

```
Switch# configure terminal
Switch(config)# ipv6 source-guard policy POL
Switch(config-sisf-sourceguard) # validate address
switch(config-sisf-sourceguard)# exit
Switch(config)# interface Po4
Switch(config)# ipv6 snooping
Switch(config-if)# ipv6 source-guard attach-policy POL
Switch(config-if)# exit
switch(config)#
```

Examples: How to attach an IPv6 Prefix Guard Policy to a Layer 2 EtherChannel Interface

The following example shows how to attach an IPv6 Prefix Guard Policy to a Layer 2 EtherChannel Interface:

```
Switch# configure terminal
Switch(config)# ipv6 source-guard policy POL
Switch (config-sisf-sourceguard)# no validate address
Switch((config-sisf-sourceguard)# validate prefix
Switch(config)# interface Po4
Switch(config-if)# ipv6 snooping
Switch(config-if)# ipv6 source-guard attach-policy POL
```

Additional References

Related Documents

Related Topic	Document Title
Implementing IPv6 Addressing and Basic Connectivity	http://www.cisco.com/US/en/products/ios/pd/6091/3/60913a.html
IPv6 network management and security topics	IPv6 Configuration Library, Cisco IOS XE Release 3SE (Catalyst 3850 Switches) http://www.cisco.com/US/en/products/ios/pd/6091/3/60913b.html
IPv6 Command Reference	IPv6 Command Reference, Cisco IOS XE Release 3SE (Catalyst 3850 Switches) http://www.cisco.com/US/en/products/ios/pd/6091/3/60913c.html

Error Message Decoder

Description	Link
To help you research and resolve system error messages in this release, use the Error Message Decoder tool.	https://www.cisco.com/cgi-bin/Support/Errordecoder/index.cgi

Technical Assistance

Description	Link
<p>The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies.</p> <p>To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds.</p> <p>Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.</p>	http://www.cisco.com/support



CHAPTER 46

Routing Information Protocol

RIP is a commonly used routing protocol in small to medium TCP/IP networks. Routing Information Protocol (RIP) is a stable protocol that uses a distance-vector algorithm to calculate routes.

This module describes how to configure RIP.

- [Prerequisites for RIP, on page 981](#)
- [Restrictions for RIP, on page 981](#)
- [Information About Routing Information Protocol, on page 981](#)
- [How to Configure Routing Information Protocol, on page 985](#)
- [Configuration Examples for Routing Information Protocol, on page 988](#)
- [Additional References for RIP, on page 989](#)
- [Feature Information for RIP, on page 989](#)

Prerequisites for RIP

You must configure `ip routing` command before you configure RIP.

Restrictions for RIP

Routing Information Protocol (RIP) uses hop count as the metric to rate the value of different routes. The hop count is the number of devices that can be traversed in a route. A directly connected network has a metric of zero; an unreachable network has a metric of 16. This limited metric range makes RIP unsuitable for large networks.

Information About Routing Information Protocol

RIP Overview

The Routing Information Protocol (RIP) Version 1 uses broadcast UDP data packets, and RIPv2 uses multicast packets to exchange the routing information. Cisco software sends routing information updates every 30 seconds, which is termed advertising. If a device does not receive an update from another device for 180

seconds or more, the receiving device marks the routes served by the nonupdating device as unusable. If there is still no update after 240 seconds, the device removes all routing table entries for the nonupdating device.

A device that is running RIP can receive a default network via an update from another device that is running RIP, or the device can source the default network using RIP. In both cases, the default network is advertised through RIP to other RIP neighbors.

The Cisco implementation of RIP Version 2 (RIPv2) supports plain text and message digest algorithm 5 (MD5) authentication, route summarization, classless interdomain routing (CIDR), and variable-length subnet masks (VLSMs).

RIP Routing Updates

The Routing Information Protocol (RIP) sends routing-update messages at regular intervals and when the network topology changes. When a device receives a RIP routing update that includes changes to an entry, the device updates its routing table to reflect the new route. The metric value for the path is increased by 1, and the sender is indicated as the next hop. RIP devices maintain only the best route (the route with the lowest metric value) to a destination. After updating its routing table, the device immediately begins transmitting RIP routing updates to inform other network devices of the change. These updates are sent independently of the regularly scheduled updates that RIP devices send.

Authentication in RIP

The Cisco implementation of the Routing Information Protocol (RIP) Version 2 (RIPv2) supports authentication, key management, route summarization, classless interdomain routing (CIDR), and variable-length subnet masks (VLSMs).

By default, the software receives RIP Version 1 (RIPv1) and RIPv2 packets, but sends only RIPv1 packets. You can configure the software to receive and send only RIPv1 packets. Alternatively, you can configure the software to receive and send only RIPv2 packets. To override the default behavior, you can configure the RIP version that an interface sends. Similarly, you can also control how packets received from an interface are processed.

RIPv1 does not support authentication. If you are sending and receiving RIP v2 packets, you can enable RIP authentication on an interface.

The key chain determines the set of keys that can be used on the interface. Authentication, including default authentication, is performed on that interface only if a key chain is configured.

Cisco supports two modes of authentication on an interface on which RIP is enabled: plain-text authentication and message digest algorithm 5 (MD5) authentication. Plain-text authentication is the default authentication in every RIPv2 packet.



Note Do not use plain text authentication in RIP packets for security purposes, because the unencrypted authentication key is sent in every RIPv2 packet. Use plain-text authentication when security is not an issue; for example, you can use plain-text authentication to ensure that misconfigured hosts do not participate in routing.

RIP Routing Metric

The Routing Information Protocol (RIP) uses a single routing metric to measure the distance between the source and the destination network. Each hop in a path from the source to the destination is assigned a hop-count value, which is typically 1. When a device receives a routing update that contains a new or changed destination network entry, the device adds 1 to the metric value indicated in the update and enters the network in the routing table. The IP address of the sender is used as the next hop. If an interface network is not specified in the routing table, it will not be advertised in any RIP update.

RIP Versions

The original version of Routing Information Protocol (RIP), is known as RIP Version 1 (RIPv1). The specification of the RIP, defined in RFC 1058, uses classful routing. Periodic routing updates do not support variable length subnet masks (VLSM) because periodic routing updates do not contain subnet information. All subnets in a network class must be of the same size. Because RIP, as per RFC 1058, does not support VLSM, it is not possible to have subnets of varying sizes inside the same network class. This limitation makes RIP vulnerable to attacks.

To rectify the deficiencies of the original RIP specification, RIP Version 2 (RIPv2), as described in RFC 2453, was developed. RIPv2 has the ability to carry subnet information; thus, it supports Classless Inter-Domain Routing (CIDR).

Exchange of Routing Information

Routing Information Protocol (RIP) is normally a broadcast protocol, and for RIP routing updates to reach nonbroadcast networks, you must configure the Cisco software to permit this exchange of routing information.

To control the set of interfaces with which you want to exchange routing updates, you can disable the sending of routing updates on specified interfaces by configuring the **passive-interface** router configuration command.

You can use an offset list to increase increasing incoming and outgoing metrics to routes learned via RIP. Optionally, you can limit the offset list with either an access list or an interface.

Routing protocols use several timers that determine variables such as the frequency of routing updates, the length of time before a route becomes invalid, and other parameters. You can adjust these timers to tune routing protocol performance to better suit your internetwork needs. You can make the following timer adjustments:

- The rate (time, in seconds, between updates) at which routing updates are sent
- The interval of time, in seconds, after which a route is declared invalid
- The interval, in seconds, during which routing information about better paths is suppressed
- The amount of time, in seconds, that must pass before a route is removed from the routing table
- The amount of time for which routing updates will be postponed

You can adjust the IP routing support in the Cisco software to enable faster convergence of various IP routing algorithms, and hence, cause quicker fallback to redundant devices. The total effect is to minimize disruptions to end users of the network in situations where quick recovery is essential

In addition, an address family can have timers that explicitly apply to that address family (or Virtual Routing and Forwarding [VRF] instance). The **timers-basic** command must be specified for an address family or the

system defaults for the **timers-basic** command are used regardless of the timer that is configured for RIP routing. The VRF does not inherit the timer values from the base RIP configuration. The VRF will always use the system default timers unless the timers are explicitly changed using the **timers-basic** command.

Neighbor Router Authentication

You can prevent your router from receiving fraudulent route updates by configuring neighbor router authentication. When configured, neighbor authentication occurs whenever routing updates are exchanged between neighbor routers. This authentication ensures that a router receives reliable routing information from a trusted source.

Without neighbor authentication, unauthorized or deliberately malicious routing updates could compromise the security of your network traffic. A security compromise could occur if an unfriendly party diverts or analyzes your network traffic. For example, an unauthorized router could send a fictitious routing update to convince your router to send traffic to an incorrect destination. This diverted traffic could be analyzed to learn confidential information about your organization or merely used to disrupt your organization's ability to effectively communicate using the network. Neighbor authentication prevents any such fraudulent route updates from being received by your router.

When neighbor authentication has been configured on a router, the router authenticates the source of each routing update packet that it receives. This is accomplished by the exchange of an authenticating key (sometimes referred to as a password) that is known to both the sending and the receiving router.

There are two types of neighbor authentication used: plain text authentication and Message Digest Algorithm Version 5 (MD5) authentication. Both forms work in the same way, with the exception that MD5 sends a "message digest" instead of the authenticating key itself. The message digest is created using the key and a message, but the key itself is not sent, preventing it from being read while it is being transmitted. Plain text authentication sends the authenticating key itself over the wire.



Note Note that plain text authentication is not recommended for use as part of your security strategy. Its primary use is to avoid accidental changes to the routing infrastructure. Using MD5 authentication, however, is a recommended security practice.

In plain text authentication, each participating neighbor router must share an authenticating key. This key is specified at each router during configuration. Multiple keys can be specified with some protocols; each key must then be identified by a key number.

In general, when a routing update is sent, the following authentication sequence occurs:

1. A router sends a routing update with a key and the corresponding key number to the neighbor router. In protocols that can have only one key, the key number is always zero. The receiving (neighbor) router checks the received key against the same key stored in its own memory.
2. If the two keys match, the receiving router accepts the routing update packet. If the two keys do not match, the routing update packet is rejected.

MD5 authentication works similarly to plain text authentication, except that the key is never sent over the wire. Instead, the router uses the MD5 algorithm to produce a "message digest" of the key (also called a "hash"). The message digest is then sent instead of the key itself. This ensures that nobody can eavesdrop on the line and learn keys during transmission.

Another form of neighbor router authentication is to configure key management using key chains. When you configure a key chain, you specify a series of keys with lifetimes, and the Cisco IOS software rotates through each of these keys. This decreases the likelihood that keys will be compromised.

How to Configure Routing Information Protocol

Enabling RIP and Configuring RIP Parameters

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **router rip**
4. **network ip-address**
5. **neighbor ip-address**
6. **auto-summary**
7. **offset-list** [*access-list-number* | *access-list-name*] {**in** | **out**} *offset* [*interface-type interface-number*]
8. **timers basic** *update invalid holddown flush* [*sleeptime*]
9. **maximum-paths** *maximum*
10. **distance** *admin-distance* [*prefix prefix-length* | *prefix-mask*]
11. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	router rip Example: Device(config)# router rip	Enables a RIP routing process and enters router configuration mode.
Step 4	network ip-address Example: Device(config-router)# network 10.1.1.0	Associates a network with a RIP routing process.

	Command or Action	Purpose
Step 5	neighbor <i>ip-address</i> Example: Device(config-router)# neighbor 10.1.1.2	Defines a neighboring device with which to exchange routing information.
Step 6	auto-summary Example: Device(config-router)# auto-summary	Restores the default behavior of automatic summarization of subnet routes into network-level routes.
Step 7	offset-list [<i>access-list-number</i> <i>access-list-name</i>] { in out } <i>offset</i> [<i>interface-type interface-number</i>] Example: Device(config-router)# offset-list 98 in 1 Ethernet 1/0	(Optional) Applies an offset list to routing metrics.
Step 8	timers basic <i>update invalid holddown flush</i> [<i>sleeptime</i>] Example: Device(config-router)# timers basic 1 2 3 4	(Optional) Adjusts routing protocol timers.
Step 9	maximum-paths <i>maximum</i> Example: Device(config-router)# maximum-paths 16	Configures the maximum number of equal cost parallel routes that RIP will install into the routing table.
Step 10	distance <i>admin-distance</i> [<i>prefix prefix-length</i> <i>prefix-mask</i>] Example: Device(config-router)# distance 85 192.168.10.0/24	Defines the administrative distance assigned to routes discovered by RIP.
Step 11	end Example: Device(config-router)# end	Exits router configuration mode and returns to privileged EXEC mode.

Specifying a RIP Version and Enabling Authentication

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **router rip**
4. **version {1 | 2}**

5. **exit**
6. **interface** *type number*
7. **ip rip send version** [1] [2]
8. **ip rip receive version** [1] [2]
9. **ip rip authentication key-chain** *name-of-chain*
10. **ip rip authentication mode** {text | md5}
11. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	router rip Example: Device(config)# router rip	Enters router configuration mode.
Step 4	version {1 2} Example: Device(config-router)# version 2	Enables the Cisco software to send only RIP Version 2 (RIPv2) packets.
Step 5	exit Example: Device(config-router)# exit	Exits the router configuration mode and enters the global configuration mode.
Step 6	interface <i>type number</i> Example: Device(config)# interface GigabitEthernet 0/0	Specifies an interface and enters interface configuration mode.
Step 7	ip rip send version [1] [2] Example: Device(config-if)# ip rip send version 2	Configures an interface to send only RIPv2 packets.
Step 8	ip rip receive version [1] [2] Example:	Configures an interface to accept only RIPv2 packets.

	Command or Action	Purpose
	Device(config-if)# ip rip receive version 2	
Step 9	ip rip authentication key-chain <i>name-of-chain</i> Example: Device(config-if)# ip rip authentication key-chain chainname	Enables RIP authentication.
Step 10	ip rip authentication mode {text md5} Example: Device(config-if)# ip rip authentication mode md5	Configures the interface to use message digest algorithm 5 (MD5) authentication (or let it default to plain-text authentication).
Step 11	end Example: Device(config-if)# end	Exits interface configuration mode and returns to privileged EXEC mode.

Configuration Examples for Routing Information Protocol

Example: Enabling RIP and Configuring RIP Parameters

```
Device> enable
Device# configure terminal
Device(config)# router rip
Device(config-router)# network 10.1.1.0
Device(config-router)# neighbor 10.1.1.2
Device(config-router)# auto-summary
Device(config-router)# offset-list 98 in 1 GigabitEthernet 1/0
Device(config-router)# timers basic 1 2 3 4
Device(config-router)# maximum-paths 16
Device(config-router)# distance 85 192.168.10.0/24
Device(config-router)# end
```

Example: Specifying a RIP Version and Enabling Authentication

```
Device> enable
Device# configure terminal
Device(config)# router rip
Device(config-router)# version 2
Device(config-router)# exit
Device(config)# interface GigabitEthernet 0/0
Device(config-if)# ip rip send version 2
Device(config-if)# ip rip receive version 2
Device(config-if)# ip rip authentication key-chain chainname
```

```
Device(config-if)# ip rip authentication mode md5
Device(config-if)# end
```

Additional References for RIP

Related Documents

Related Topic	Document Title
IP Routing: RIP commands	Cisco IOS IP Routing: RIP Command Reference

Standards and RFCs

Standards/RFC	Title
RFC 1058	<i>Routing Information Protocol</i>
RFC 2453	<i>RIP Version 2</i>

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

Feature Information for RIP

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Table 103: Feature Information for RIP

Feature Name	Releases	Feature Information
RIP (Routing Information Protocol)	Cisco IOS Release 15.2(5)E2	RIP is a commonly used routing protocol in small to medium TCP/IP networks. RIP is a stable protocol that uses a distance-vector algorithm to calculate routes.



PART **XI**

Security

- [Security Features Overview, on page 993](#)
- [Preventing Unauthorized Access , on page 997](#)
- [Controlling Switch Access with Passwords and Privilege Levels , on page 999](#)
- [Configuring TACACS+, on page 1015](#)
- [Configuring RADIUS, on page 1059](#)
- [RADIUS Server Load Balancing, on page 1101](#)
- [RADIUS Change of Authorization Support, on page 1117](#)
- [Local AAA Server, on page 1133](#)
- [Configuring Kerberos , on page 1143](#)
- [Configuring Accounting, on page 1149](#)
- [Configuring Local Authentication and Authorization , on page 1179](#)
- [Certification Authority Interoperability, on page 1183](#)
- [MAC Authentication Bypass, on page 1201](#)
- [Password Strength and Management for Common Criteria, on page 1211](#)
- [AAA-SERVER-MIB Set Operation, on page 1219](#)
- [Configuring Secure Shell, on page 1225](#)
- [Secure Shell Version 2 Support, on page 1245](#)
- [X.509v3 Certificates for SSH Authentication, on page 1271](#)
- [Configuring Secure Socket Layer HTTP, on page 1283](#)
- [Access Control List Overview, on page 1297](#)
- [Configuring IPv4 Access Control Lists, on page 1307](#)
- [IPv6 Access Control Lists, on page 1349](#)
- [ACL Support for Filtering IP Options, on page 1367](#)
- [VLAN Access Control Lists, on page 1377](#)

- [Configuring DHCP](#) , on page 1397
- [Configuring IP Source Guard](#) , on page 1421
- [Configuring Dynamic ARP Inspection](#), on page 1429
- [Configuring IEEE 802.1x Port-Based Authentication](#), on page 1447
- [Configuring Web-Based Authentication](#), on page 1539
- [Auto Identity](#), on page 1573
- [Configuring Port-Based Traffic Control](#), on page 1585
- [Configuring Cisco TrustSec](#), on page 1617
- [Configuring FIPS](#), on page 1619
- [Configuring Control Plane Policing](#), on page 1621



CHAPTER 47

Security Features Overview

- [Security Features Overview](#), on page 993

Security Features Overview

The switch supports a LAN base image or a LAN lite image with a reduced feature set, depending on switch hardware. The security features are as follows:

- IPv6 First Hop Security—A suite of security features to be applied at the first hop switch to protect against vulnerabilities inherent in IPv6 networks. These include, Binding Integrity Guard (Binding Table), Router Advertisement Guard (RA Guard), DHCP Guard, IPv6 Neighbor Discovery Inspection (ND Guard).
- Web Authentication—Allows a supplicant (client) that does not support IEEE 802.1x functionality to be authenticated using a web browser.
- Local Web Authentication Banner—A custom banner or an image file displayed at a web authentication login screen.
- IEEE 802.1x Authentication with ACLs and the RADIUS Filter-Id Attribute
- Password-protected access (read-only and read-write access) to management interfaces (device manager, Network Assistant, and the CLI) for protection against unauthorized configuration changes
- Multilevel security for a choice of security level, notification, and resulting actions
- Static MAC addressing for ensuring security
- Protected port option for restricting the forwarding of traffic to designated ports on the same switch
- Port security option for limiting and identifying MAC addresses of the stations allowed to access the port
- VLAN aware port security option to shut down the VLAN on the port when a violation occurs, instead of shutting down the entire port.
- Port security aging to set the aging time for secure addresses on a port.
- Protocol storm protection to control the rate of incoming protocol traffic to a switch by dropping packets that exceed a specified ingress rate.
- BPDU guard for shutting down a Port Fast-configured port when an invalid configuration occurs.

- Standard and extended IP access control lists (ACLs) for defining inbound security policies on Layer 2 interfaces (port ACLs).
- Extended MAC access control lists for defining security policies in the inbound direction on Layer 2 interfaces.
- Source and destination MAC-based ACLs for filtering non-IP traffic.
- DHCP snooping to filter untrusted DHCP messages between untrusted hosts and DHCP servers.
- IP source guard to restrict traffic on nonrouted interfaces by filtering traffic based on the DHCP snooping database and IP source bindings.
- Dynamic ARP inspection to prevent malicious attacks on the switch by not relaying invalid ARP requests and responses to other ports in the same VLAN.
- IEEE 802.1x port-based authentication to prevent unauthorized devices (clients) from gaining access to the network. These 802.1x features are supported:
 - Support for single-host, multi-host, multi-auth, and multi-domain-auth modes.
 - Multidomain authentication (MDA) to allow both a data device and a voice device, such as an IP phone (Cisco or non-Cisco), to independently authenticate on the same IEEE 802.1x-enabled switch port.
 - Dynamic voice virtual LAN (VLAN) for MDA to allow a dynamic voice VLAN on an MDA-enabled port.
 - VLAN assignment for restricting 802.1x-authenticated users to a specified VLAN.
 - Support for VLAN assignment on a port configured for multi-auth mode. The RADIUS server assigns a VLAN to the first host to authenticate on the port, and subsequent hosts use the same VLAN. Voice VLAN assignment is supported for one IP phone.
 - Port security for controlling access to 802.1x ports.
 - Voice VLAN to permit a Cisco IP Phone to access the voice VLAN regardless of the authorized or unauthorized state of the port.
 - IP phone detection enhancement to detect and recognize a Cisco IP phone.
 - Guest VLAN to provide limited services to non-802.1x-compliant users.
 - Restricted VLAN to provide limited services to users who are 802.1x compliant, but do not have the credentials to authenticate via the standard 802.1x processes.
 - 802.1x accounting to track network usage.
 - 802.1x with wake-on-LAN to allow dormant PCs to be powered on based on the receipt of a specific Ethernet frame.
 - 802.1x readiness check to determine the readiness of connected end hosts before configuring IEEE 802.1x on the switch.
 - Voice aware 802.1x security to apply traffic violation actions only on the VLAN on which a security violation occurs.
 - MAC authentication bypass (MAB) to authorize clients based on the client MAC address.

- Network Admission Control (NAC) Layer 2 802.1x validation of the antivirus condition or posture of endpoint systems or clients before granting the devices network access.



Note NAC is not supported on LanLite images.

- Network Edge Access Topology (NEAT) with 802.1X switch supplicant, host authorization with CISP, and auto enablement to authenticate a switch outside a wiring closet as a supplicant to another switch.



Note NEAT is not supported on LanLite images.

- IEEE 802.1x with open access to allow a host to access the network before being authenticated.



Note This feature is not supported on LanLite images.

- IEEE 802.1x authentication with downloadable ACLs and redirect URLs to allow per-user ACL downloads from a Cisco Secure ACS server to an authenticated switch.
- Support for dynamic creation or attachment of an auth-default ACL on a port that has no configured static ACLs.



Note This feature is not supported on LanLite images.

- Flexible-authentication sequencing to configure the order of the authentication methods that a port tries when authenticating a new host.
- Multiple-user authentication to allow more than one host to authenticate on an 802.1x-enabled port.
- TACACS+, a proprietary feature for managing network security through a TACACS server for both IPv4 and IPv6.
- RADIUS for verifying the identity of, granting access to, and tracking the actions of remote users through authentication, authorization, and accounting (AAA) services for both IPv4 and IPv6.
- Enhancements to RADIUS, TACACS+, and SSH to function over IPv6.
- Secure Socket Layer (SSL) Version 3.0 support for the HTTP 1.1 server authentication, encryption, and message integrity and HTTP client authentication to allow secure HTTP communications (requires the cryptographic version of the software).
- IEEE 802.1x Authentication with ACLs and the RADIUS Filter-Id Attribute.
- Support for IP source guard on static hosts.
- RADIUS Change of Authorization (CoA) to change the attributes of a certain session after it is authenticated. When there is a change in policy for a user or user group in AAA, administrators can send

the RADIUS CoA packets from the AAA server, such as Cisco Identity Services Engine, or Cisco Secure ACS to reinitialize authentication, and apply to the new policies.

- IEEE 802.1x User Distribution to allow deployments with multiple VLANs (for a group of users) to improve scalability of the network by load balancing users across different VLANs. Authorized users are assigned to the least populated VLAN in the group, assigned by RADIUS server.



Note This feature is not supported on LanLite images.

- Support for critical VLAN—multi-host/multi-auth enabled ports are placed in a critical VLAN in order to permit access to critical resources if AAA server becomes unreachable.



Note This feature is not supported on LanLite images.

- Support for Network Edge Access Topology (NEAT) to change the port host mode and to apply a standard port configuration on the authenticator switch port.
- VLAN-ID based MAC authentication to use the combined VLAN and MAC address information for user authentication to prevent network access from unauthorized VLANs.
- MAC move to allow hosts (including the hosts connected behind an IP phone) to move across ports within the same switch without any restrictions to enable mobility. With MAC move, the switch treats the reappearance of the same MAC address on another port in the same way as a completely new MAC address.
- Support for 3DES and AES with version 3 of the Simple Network Management Protocol (SNMPv3). This release adds support for the 168-bit Triple Data Encryption Standard (3DES) and the 128-bit, 192-bit, and 256-bit Advanced Encryption Standard (AES) encryption algorithms to SNMPv3.
- Support for Cisco TrustSec SXP protocol. This feature is not supported on LanLite images.



CHAPTER 48

Preventing Unauthorized Access

- [Preventing Unauthorized Access, on page 997](#)

Preventing Unauthorized Access

You can prevent unauthorized users from reconfiguring your switch and viewing configuration information. Typically, you want network administrators to have access to your switch while you restrict access to users who dial from outside the network through an asynchronous port, connect from outside the network through a serial port, or connect through a terminal or workstation from within the local network.

To prevent unauthorized access into your switch, you should configure one or more of these security features:

- At a minimum, you should configure passwords and privileges at each switch port. These passwords are locally stored on the switch. When users attempt to access the switch through a port or line, they must enter the password specified for the port or line before they can access the switch.
- For an additional layer of security, you can also configure username and password pairs, which are locally stored on the switch. These pairs are assigned to lines or ports and authenticate each user before that user can access the switch. If you have defined privilege levels, you can also assign a specific privilege level (with associated rights and privileges) to each username and password pair.
- If you want to use username and password pairs, but you want to store them centrally on a server instead of locally, you can store them in a database on a security server. Multiple networking devices can then use the same database to obtain user authentication (and, if necessary, authorization) information.
- You can also enable the login enhancements feature, which logs both failed and unsuccessful login attempts. Login enhancements can also be configured to block future login attempts after a set number of unsuccessful attempts are made. For more information, see the Cisco IOS Login Enhancements documentation.



CHAPTER 49

Controlling Switch Access with Passwords and Privilege Levels

- [Restrictions for Controlling Switch Access with Passwords and Privileges, on page 999](#)
- [Information About Passwords and Privilege Levels, on page 999](#)
- [How to Control Switch Access with Passwords and Privilege Levels, on page 1002](#)
- [Monitoring Switch Access, on page 1012](#)
- [Configuration Examples for Setting Passwords and Privilege Levels, on page 1013](#)
- [Additional References, on page 1013](#)

Restrictions for Controlling Switch Access with Passwords and Privileges

The following are the restrictions for controlling switch access with passwords and privileges:

- Disabling password recovery will not work if you have set the switch to boot up manually by using the **boot manual** global configuration command. This command produces the boot loader prompt (*switch:*) after the switch is power cycled.

Information About Passwords and Privilege Levels

Default Password and Privilege Level Configuration

A simple way of providing terminal access control in your network is to use passwords and assign privilege levels. Password protection restricts access to a network or network device. Privilege levels define what commands users can enter after they have logged into a network device.

This table shows the default password and privilege level configuration.

Table 104: Default Password and Privilege Levels

Feature	Default Setting
Enable password and privilege level	No password is defined. The default is level 15 (privileged EXEC level). The password is not encrypted in the configuration file.
Enable secret password and privilege level	No password is defined. The default is level 15 (privileged EXEC level). The password is encrypted before it is written to the configuration file.
Line password	No password is defined.

Additional Password Security

To provide an additional layer of security, particularly for passwords that cross the network or that are stored on a Trivial File Transfer Protocol (TFTP) server, you can use either the **enable password** or **enable secret** global configuration commands. Both commands accomplish the same thing; that is, you can establish an encrypted password that users must enter to access privileged EXEC mode (the default) or any privilege level you specify.

We recommend that you use the **enable secret** command because it uses an improved encryption algorithm.

If you configure the **enable secret** command, it takes precedence over the **enable password** command; the two commands cannot be in effect simultaneously.

If you enable password encryption, it applies to all passwords including username passwords, authentication key passwords, the privileged command password, and console and virtual terminal line passwords.

Password Recovery

By default, any end user with physical access to the switch can recover from a lost password by interrupting the boot process while the switch is powering on and then by entering a new password.

The password-recovery disable feature protects access to the switch password by disabling part of this functionality. When this feature is enabled, the end user can interrupt the boot process only by agreeing to set the system back to the default configuration. With password recovery disabled, you can still interrupt the boot process and change the password, but the configuration file (config.text) and the VLAN database file (vlan.dat) are deleted.

If you disable password recovery, we recommend that you keep a backup copy of the configuration file on a secure server in case the end user interrupts the boot process and sets the system back to default values. Do not keep a backup copy of the configuration file on the switch. If the switch is operating in VTP transparent mode, we recommend that you also keep a backup copy of the VLAN database file on a secure server. When the switch is returned to the default system configuration, you can download the saved files to the switch by using the Xmodem protocol.

Terminal Line Telnet Configuration

When you power-up your switch for the first time, an automatic setup program runs to assign IP information and to create a default configuration for continued use. The setup program also prompts you to configure your switch for Telnet access through a password. If you did not configure this password during the setup program, you can configure it when you set a Telnet password for a terminal line.

Username and Password Pairs

You can configure username and password pairs, which are locally stored on the switch. These pairs are assigned to lines or ports and authenticate each user before that user can access the switch. If you have defined privilege levels, you can also assign a specific privilege level (with associated rights and privileges) to each username and password pair.

Privilege Levels

Cisco devices use privilege levels to provide password security for different levels of switch operation. By default, the Cisco IOS software operates in two modes (privilege levels) of password security: user EXEC (Level 1) and privileged EXEC (Level 15). You can configure up to 16 hierarchical levels of commands for each mode. By configuring multiple passwords, you can allow different sets of users to have access to specified commands.

Privilege Levels on Lines

Users can override the privilege level you set using the **privilege level** line configuration command by logging in to the line and enabling a different privilege level. They can lower the privilege level by using the **disable** command. If users know the password to a higher privilege level, they can use that password to enable the higher privilege level. You might specify a high level or privilege level for your console line to restrict line usage.

For example, if you want many users to have access to the **clear line** command, you can assign it level 2 security and distribute the level 2 password fairly widely. But if you want more restricted access to the **configure** command, you can assign it level 3 security and distribute that password to a more restricted group of users.

Command Privilege Levels

When you set a command to a privilege level, all commands whose syntax is a subset of that command are also set to that level. For example, if you set the **show ip traffic** command to level 15, the **show** commands and **show ip** commands are automatically set to privilege level 15 unless you set them individually to different levels.

How to Control Switch Access with Passwords and Privilege Levels

Setting or Changing a Static Enable Password

The enable password controls access to the privileged EXEC mode. Follow these steps to set or change a static enable password:

SUMMARY STEPS

1. `enable`
2. `configure terminal`
3. `enable password password`
4. `end`
5. `show running-config`
6. `copy running-config startup-config`

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> <code>enable</code>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode.
Step 3	enable password <i>password</i> Example: Device(config)# <code>enable password secret321</code>	Defines a new password or changes an existing password for access to privileged EXEC mode. By default, no password is defined. For <i>password</i> , specify a string from 1 to 25 alphanumeric characters. The string cannot start with a number, is case sensitive, and allows spaces but ignores leading spaces. It can contain the question mark (?) character if you precede the question mark with the key combination Ctrl-v when you create the password; for example, to create the password abc?123, do this: <ol style="list-style-type: none"> a. Enter <code>abc</code>. b. Enter <code>Ctrl-v</code>.

	Command or Action	Purpose
		<p>c. Enter ?123.</p> <p>When the system prompts you to enter the enable password, you need not precede the question mark with the Ctrl-v; you can simply enter abc?123 at the password prompt.</p>
Step 4	<p>end</p> <p>Example:</p> <pre>Device(config)# end</pre>	Returns to privileged EXEC mode.
Step 5	<p>show running-config</p> <p>Example:</p> <pre>Device# show running-config</pre>	Verifies your entries.
Step 6	<p>copy running-config startup-config</p> <p>Example:</p> <pre>Device# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Protecting Enable and Enable Secret Passwords with Encryption

Follow these steps to establish an encrypted password that users must enter to access privileged EXEC mode (the default) or any privilege level you specify:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. Use one of the following:
 - `enable password [level level] {password encryption-type encrypted-password}`
 - `enable secret [level level] {password encryption-type encrypted-password}`
4. **service password-encryption**
5. **end**
6. **show running-config**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	Use one of the following: <ul style="list-style-type: none"> • <code>enable password [level level] {password encryption-type encrypted-password}</code> • <code>enable secret [level level] {password encryption-type encrypted-password}</code> Example: Device(config)# enable password example102 or Device(config)# enable secret level 1 password secret123sample	<ul style="list-style-type: none"> • Defines a new password or changes an existing password for access to privileged EXEC mode. • Defines a secret password, which is saved using a nonreversible encryption method. <ul style="list-style-type: none"> • (Optional) For <i>level</i>, the range is from 0 to 15. Level 1 is normal user EXEC mode privileges. The default level is 15 (privileged EXEC mode privileges). • For <i>password</i>, specify a string from 1 to 25 alphanumeric characters. The string cannot start with a number, is case sensitive, and allows spaces but ignores leading spaces. By default, no password is defined. • (Optional) For <i>encryption-type</i>, only type 5, a Cisco proprietary encryption algorithm, is available. If you specify an encryption type, you must provide an encrypted password—an encrypted password that you copy from another switch configuration. <p>Note If you specify an encryption type and then enter a clear text password, you can not re-enter privileged EXEC mode. You cannot recover a lost encrypted password by any method.</p>
Step 4	service password-encryption Example: Device(config)# service password-encryption	(Optional) Encrypts the password when the password is defined or when the configuration is written. Encryption prevents the password from being readable in the configuration file.

	Command or Action	Purpose
Step 5	end Example: <pre>Device(config)# end</pre>	Returns to privileged EXEC mode.
Step 6	show running-config Example: <pre>Device# show running-config</pre>	Verifies your entries.
Step 7	copy running-config startup-config Example: <pre>Device# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Disabling Password Recovery

Follow these steps to disable password recovery to protect the security of your switch:

Before you begin

If you disable password recovery, we recommend that you keep a backup copy of the configuration file on a secure server in case the end user interrupts the boot process and sets the system back to default values. Do not keep a backup copy of the configuration file on the switch. If the switch is operating in VTP transparent mode, we recommend that you also keep a backup copy of the VLAN database file on a secure server. When the switch is returned to the default system configuration, you can download the saved files to the switch by using the Xmodem protocol.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **system disable password recovery switch** {all | <1-9>}
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Device> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.

	Command or Action	Purpose
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	system disable password recovery switch {all <1-9>} Example: Device(config)# system disable password recovery switch all	Disables password recovery. <ul style="list-style-type: none"> • <i>all</i> - Sets the configuration on switches in stack. • <i><1-9></i> - Sets the configuration on the Switch Number selected. This setting is saved in an area of the flash memory that is accessible by the boot loader and the Cisco IOS image, but it is not part of the file system and is not accessible by any user.
Step 4	end Example: Device(config)# end	Returns to privileged EXEC mode.

What to do next

To remove **disable password recovery**, use the **no system disable password recovery switch all** global configuration command.

Setting a Telnet Password for a Terminal Line

Beginning in user EXEC mode, follow these steps to set a Telnet password for the connected terminal line:

Before you begin

- Attach a PC or workstation with emulation software to the switch console port, or attach a PC to the Ethernet management port.
- The default data characteristics of the console port are 9600, 8, 1, no parity. You might need to press the Return key several times to see the command-line prompt.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **line vty 0 15**
4. **password *password***
5. **end**
6. **show running-config**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Device> enable</pre>	Note If a password is required for access to privileged EXEC mode, you will be prompted for it. Enters privileged EXEC mode.
Step 2	configure terminal Example: <pre>Device# configure terminal</pre>	Enters global configuration mode.
Step 3	line vty 0 15 Example: <pre>Device(config)# line vty 0 15</pre>	Configures the number of Telnet sessions (lines), and enters line configuration mode. There are 16 possible sessions on a command-capable Device. The 0 and 15 mean that you are configuring all 16 possible Telnet sessions.
Step 4	password <i>password</i> Example: <pre>Device(config-line)# password abcxyz543</pre>	Sets a Telnet password for the line or lines. For <i>password</i> , specify a string from 1 to 25 alphanumeric characters. The string cannot start with a number, is case sensitive, and allows spaces but ignores leading spaces. By default, no password is defined.
Step 5	end Example: <pre>Device(config-line)# end</pre>	Returns to privileged EXEC mode.
Step 6	show running-config Example: <pre>Device# show running-config</pre>	Verifies your entries.
Step 7	copy running-config startup-config Example: <pre>Device# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Configuring Username and Password Pairs

Follow these steps to configure username and password pairs:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **username** *name* [**privilege** *level*] {**password** *encryption-type password*}
4. Use one of the following:
 - **line console 0**
 - **line vty 0 15**
5. **login local**
6. **end**
7. **show running-config**
8. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	username <i>name</i> [privilege <i>level</i>] { password <i>encryption-type password</i> } Example: Device(config)# username adamsample privilege 1 password secret456 Device(config)# username 111111111111 mac attribute	Sets the username, privilege level, and password for each user. <ul style="list-style-type: none"> • For <i>name</i>, specify the user ID as one word or the MAC address. Spaces and quotation marks are not allowed. • You can configure a maximum of 12000 clients each, for both username and MAC filter. • (Optional) For <i>level</i>, specify the privilege level the user has after gaining access. The range is 0 to 15. Level 15 gives privileged EXEC mode access. Level 1 gives user EXEC mode access. • For <i>encryption-type</i>, enter 0 to specify that an unencrypted password will follow. Enter 7 to specify that a hidden password will follow. • For <i>password</i>, specify the password the user must enter to gain access to the Device. The password must be from 1 to 25 characters, can contain embedded spaces, and must be the last option specified in the username command.

	Command or Action	Purpose
Step 4	Use one of the following: <ul style="list-style-type: none"> • line console 0 • line vty 0 15 Example: <pre>Device(config)# line console 0</pre> or <pre>Device(config)# line vty 15</pre>	Enters line configuration mode, and configures the console port (line 0) or the VTY lines (line 0 to 15).
Step 5	login local Example: <pre>Device(config-line)# login local</pre>	Enables local password checking at login time. Authentication is based on the username specified in Step 3.
Step 6	end Example: <pre>Device(config)# end</pre>	Returns to privileged EXEC mode.
Step 7	show running-config Example: <pre>Device# show running-config</pre>	Verifies your entries.
Step 8	copy running-config startup-config Example: <pre>Device# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Setting the Privilege Level for a Command

Follow these steps to set the privilege level for a command:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **privilege mode level level command**
4. **enable password level level password**
5. **end**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Device> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: <pre>Device# configure terminal</pre>	Enters global configuration mode.
Step 3	privilege mode level level command Example: <pre>Device(config)# privilege exec level 14 configure</pre>	Sets the privilege level for a command. <ul style="list-style-type: none"> • For <i>mode</i>, enter configure for global configuration mode, exec for EXEC mode, interface for interface configuration mode, or line for line configuration mode. • For <i>level</i>, the range is from 0 to 15. Level 1 is for normal user EXEC mode privileges. Level 15 is the level of access permitted by the enable password. • For <i>command</i>, specify the command to which you want to restrict access.
Step 4	enable password level level password Example: <pre>Device(config)# enable password level 14 SecretPswd14</pre>	Specifies the password to enable the privilege level. <ul style="list-style-type: none"> • For <i>level</i>, the range is from 0 to 15. Level 1 is for normal user EXEC mode privileges. • For <i>password</i>, specify a string from 1 to 25 alphanumeric characters. The string cannot start with a number, is case sensitive, and allows spaces but ignores leading spaces. By default, no password is defined.
Step 5	end Example: <pre>Device(config)# end</pre>	Returns to privileged EXEC mode.
Step 6	copy running-config startup-config Example: <pre>Device# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Changing the Default Privilege Level for Lines

Follow these steps to change the default privilege level for the specified line:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **line vty *line***
4. **privilege level *level***
5. **end**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	line vty <i>line</i> Example: Device(config)# line vty 10	Selects the virtual terminal line on which to restrict access.
Step 4	privilege level <i>level</i> Example: Device(config)# privilege level 15	Changes the default privilege level for the line. For <i>level</i> , the range is from 0 to 15. Level 1 is for normal user EXEC mode privileges. Level 15 is the level of access permitted by the enable password.
Step 5	end Example: Device(config)# end	Returns to privileged EXEC mode.
Step 6	copy running-config startup-config Example:	(Optional) Saves your entries in the configuration file.

	Command or Action	Purpose
	Device# <code>copy running-config startup-config</code>	

What to do next

Users can override the privilege level you set using the **privilege level** line configuration command by logging in to the line and enabling a different privilege level. They can lower the privilege level by using the **disable** command. If users know the password to a higher privilege level, they can use that password to enable the higher privilege level. You might specify a high level or privilege level for your console line to restrict line usage.

Logging into and Exiting a Privilege Level

Beginning in user EXEC mode, follow these steps to log into a specified privilege level and exit a specified privilege level.

SUMMARY STEPS

1. **enable** *level*
2. **disable** *level*

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable <i>level</i> Example: Device> <code>enable 15</code>	Logs in to a specified privilege level. Following the example, Level 15 is privileged EXEC mode. For <i>level</i> , the range is 0 to 15.
Step 2	disable <i>level</i> Example: Device# <code>disable 1</code>	Exits to a specified privilege level. Following the example, Level 1 is user EXEC mode. For <i>level</i> , the range is 0 to 15.

Monitoring Switch Access

Table 105: Commands for Displaying DHCP Information

<code>show privilege</code>	Displays the privilege level configuration.
-----------------------------	---

Configuration Examples for Setting Passwords and Privilege Levels

Example: Setting or Changing a Static Enable Password

This example shows how to change the enable password to *11u2c3k4y5*. The password is not encrypted and provides access to level 15 (traditional privileged EXEC mode access):

```
Device(config)# enable password 11u2c3k4y5
```

Example: Protecting Enable and Enable Secret Passwords with Encryption

This example shows how to configure the encrypted password *\$1\$FaD0\$Xyti5Rkls3LoyxzS8* for privilege level 2:

```
Device(config)# enable secret level 2 5 $1$FaD0$Xyti5Rkls3LoyxzS8
```

Example: Setting a Telnet Password for a Terminal Line

This example shows how to set the Telnet password to *let45me67in89*:

```
Device(config)# line vty 10
Device(config-line)# password let45me67in89
```

Example: Setting the Privilege Level for a Command

This example shows how to set the **configure** command to privilege level 14 and define *SecretPswd14* as the password users must enter to use level 14 commands:

```
Device(config)# privilege exec level 14 configure
Device(config)# enable password level 14 SecretPswd14
```

Additional References

Error Message Decoder

Description	Link
To help you research and resolve system error messages in this release, use the Error Message Decoder tool.	https://www.cisco.com/cgi-bin/Support/Errordecoder/index.cgi

MIBs

MB	MIBs Link
	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
<p>The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies.</p> <p>To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds.</p> <p>Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.</p>	http://www.cisco.com/support



CHAPTER 50

Configuring TACACS+

TACACS+ is a security application that provides centralized validation of users attempting to gain access to a router or network access server. TACACS+ provides detailed accounting information and flexible administrative control over authentication and authorization processes. TACACS+ is facilitated through authentication, authorization and accounting (AAA) and can be enabled only through AAA commands.

- [Finding Feature Information, on page 1015](#)
- [Prerequisites for TACACS+, on page 1015](#)
- [Restrictions for TACACS+, on page 1016](#)
- [Information About TACACS+, on page 1017](#)
- [How to Configure TACACS+, on page 1041](#)
- [Configuration Examples for TACACS+, on page 1052](#)
- [Additional References for TACACS+, on page 1056](#)
- [Feature Information for TACACS+, on page 1056](#)

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to <http://www.cisco.com/go/cfn>. An account on Cisco.com is not required.

Prerequisites for TACACS+

The following are the prerequisites for set up and configuration of switch access with TACACS+ (must be performed in the order presented):

1. Configure the switches with the TACACS+ server addresses.
2. Set an authentication key.
3. Configure the key from Step 2 on the TACACS+ servers.
4. Enable authentication, authorization, and accounting (AAA).

5. Create a login authentication method list.
6. Apply the list to the terminal lines.
7. Create an authorization and accounting method list.

The following are the prerequisites for controlling switch access with TACACS+:

- You must have access to a configured TACACS+ server to configure TACACS+ features on your switch. Also, you must have access to TACACS+ services maintained in a database on a TACACS+ daemon typically running on a LINUX or Windows workstation.
- We recommend a redundant connection between a switch stack and the TACACS+ server. This is to help ensure that the TACACS+ server remains accessible in case one of the connected stack members is removed from the switch stack.
- You need a system running the TACACS+ daemon software to use TACACS+ on your switch.
- To use TACACS+, it must be enabled.
- Authorization must be enabled on the switch to be used.
- Users must first successfully complete TACACS+ authentication before proceeding to TACACS+ authorization.
- To use any of the AAA commands listed in this section or elsewhere, you must first enable AAA with the **aaa new-model** command.
- At a minimum, you must identify the host or hosts maintaining the TACACS+ daemon and define the method lists for TACACS+ authentication. You can optionally define method lists for TACACS+ authorization and accounting.
- The method list defines the types of authentication to be performed and the sequence in which they are performed; it must be applied to a specific port before any of the defined authentication methods are performed. The only exception is the default method list (which, by coincidence, is named *default*). The default method list is automatically applied to all ports except those that have a named method list explicitly defined. A defined method list overrides the default method list.
- Use TACACS+ for privileged EXEC access authorization if authentication was performed by using TACACS+.
- Use the local database if authentication was not performed by using TACACS+.

Restrictions for TACACS+

TACACS+ can be enabled only through AAA commands.

Information About TACACS+

TACACS+ and Switch Access

This section describes TACACS+. TACACS+ provides detailed accounting information and flexible administrative control over the authentication and authorization processes. It is facilitated through authentication, authorization, accounting (AAA) and can be enabled only through AAA commands.

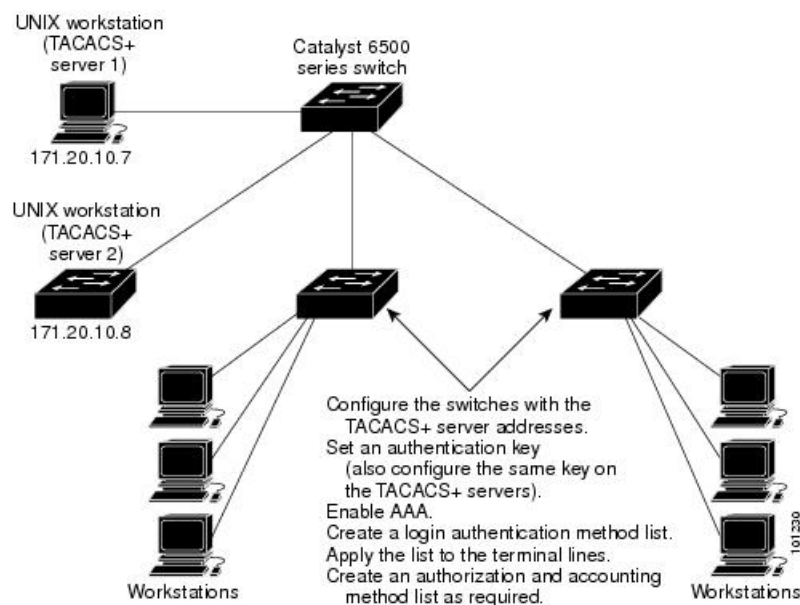
TACACS+ Overview

TACACS+ is a security application that provides centralized validation of users attempting to gain access to your switch.

TACACS+ provides for separate and modular authentication, authorization, and accounting facilities. TACACS+ allows for a single access control server (the TACACS+ daemon) to provide each service—authentication, authorization, and accounting—independently. Each service can be tied into its own database to take advantage of other services available on that server or on the network, depending on the capabilities of the daemon.

The goal of TACACS+ is to provide a method for managing multiple network access points from a single management service. Your switch can be a network access server along with other Cisco routers and access servers.

Figure 85: Typical TACACS+ Network Configuration



TACACS+, administered through the AAA security services, can provide these services:

- Authentication—Provides complete control of authentication through login and password dialog, challenge and response, and messaging support.

The authentication facility can conduct a dialog with the user (for example, after a username and password are provided, to challenge a user with several questions, such as home address, mother's maiden name,

service type, and social security number). The TACACS+ authentication service can also send messages to user screens. For example, a message could notify users that their passwords must be changed because of the company's password aging policy.

- **Authorization**—Provides fine-grained control over user capabilities for the duration of the user's session, including but not limited to setting autocommands, access control, session duration, or protocol support. You can also enforce restrictions on what commands a user can execute with the TACACS+ authorization feature.
- **Accounting**—Collects and sends information used for billing, auditing, and reporting to the TACACS+ daemon. Network managers can use the accounting facility to track user activity for a security audit or to provide information for user billing. Accounting records include user identities, start and stop times, executed commands (such as PPP), number of packets, and number of bytes.

The TACACS+ protocol provides authentication between the switch and the TACACS+ daemon, and it ensures confidentiality because all protocol exchanges between the switch and the TACACS+ daemon are encrypted.

TACACS+ Operation

When a user attempts a simple ASCII login by authenticating to a switch using TACACS+, this process occurs:

1. When the connection is established, the switch contacts the TACACS+ daemon to obtain a username prompt to show to the user. The user enters a username, and the switch then contacts the TACACS+ daemon to obtain a password prompt. The switch displays the password prompt to the user, the user enters a password, and the password is then sent to the TACACS+ daemon.

TACACS+ allows a dialog between the daemon and the user until the daemon receives enough information to authenticate the user. The daemon prompts for a username and password combination, but can include other items, such as the user's mother's maiden name.

2. The switch eventually receives one of these responses from the TACACS+ daemon:
 - **ACCEPT**—The user is authenticated and service can begin. If the switch is configured to require authorization, authorization begins at this time.
 - **REJECT**—The user is not authenticated. The user can be denied access or is prompted to retry the login sequence, depending on the TACACS+ daemon.
 - **ERROR**—An error occurred at some time during authentication with the daemon or in the network connection between the daemon and the switch. If an ERROR response is received, the switch typically tries to use an alternative method for authenticating the user.
 - **CONTINUE**—The user is prompted for additional authentication information.

After authentication, the user undergoes an additional authorization phase if authorization has been enabled on the switch. Users must first successfully complete TACACS+ authentication before proceeding to TACACS+ authorization.

3. If TACACS+ authorization is required, the TACACS+ daemon is again contacted, and it returns an ACCEPT or REJECT authorization response. If an ACCEPT response is returned, the response contains data in the form of attributes that direct the EXEC or NETWORK session for that user and the services that the user can access:
 - Telnet, Secure Shell (SSH), rlogin, or privileged EXEC services

- Connection parameters, including the host or client IP address, access list, and user timeouts

Method List

A method list defines the sequence and methods to be used to authenticate, to authorize, or to keep accounts on a user. You can use method lists to designate one or more security protocols to be used, thus ensuring a backup system if the initial method fails. The software uses the first method listed to authenticate, to authorize, or to keep accounts on users; if that method does not respond, the software selects the next method in the list. This process continues until there is successful communication with a listed method or the method list is exhausted.

If a method list is configured under VTY lines, the corresponding method list must be added to AAA. The following example shows how to configure a method list under a VTY line:

```
Device# configure terminal
Device(config)# line vty 0 4
Device(config)# authorization commands 15 auth1
```

The following example shows how to configure a method list in AAA:

```
Device# configure terminal
Device(config)# aaa new-model
Device(config)# aaa authorization commands 15 auth1 group tacacs+
```

If no method list is configured under VTY lines, the default method list must be added to AAA. The following example shows a VTY configuration without a method list:

```
Device# configure terminal
Device(config)# line vty 0 4
```

The following example shows how to configure the default method list:

```
Device# configure terminal
Device(config)# aaa new-model
Device(config)# aaa authorization commands 15 default group tacacs+
```

TACACS AV Pairs

The network access server implements TACACS+ authorization and accounting functions by transmitting and receiving TACACS+ attribute-value (AV) pairs for each user session.

TACACS Authentication and Authorization AV Pairs

The following table lists and describes the supported TACACS+ authentication and authorization AV pairs and specifies the Cisco IOS release in which they are implemented.

Table 106: Supported TACACS+ Authentication and Authorization AV Pairs

Attribute	Description	11.0	11.1	11.2	11.3	12.0	12.1	12.2
acl=x	ASCII number representing a connection access list. Used only when service=shell.	yes	yes	yes	yes	yes	yes	yes

Attribute	Description	11.0	11.1	11.2	11.3	12.0	12.1	12.2
addr=x	A network address. Used with service=slip, service=ppp, and protocol=ip. Contains the IP address that the remote host should use when connecting via SLIP or PPP/IP. For example, addr=10.2.3.4.	yes	yes	yes	yes	yes	yes	yes
addr-pool=x	Specifies the name of a local pool from which to get the address of the remote host. Used with service=ppp and protocol=ip. Note that addr-pool works in conjunction with local pooling. It specifies the name of a local pool (which must be preconfigured on the network access server). Use the ip-local pool command to declare local pools. For example: ip address-pool local ip local pool boo 10.0.0.1 10.0.0.10 ip local pool moo 10.0.0.1 10.0.0.20 You can then use TACACS+ to return addr-pool=boo or addr-pool=moo to indicate the address pool from which you want to get this remote node's address.	yes	yes	yes	yes	yes	yes	yes
autocmd=x	Specifies an autocommand to be executed at EXEC startup (for example, autocmd=telnet example.com). Used only with service=shell.	yes	yes	yes	yes	yes	yes	yes
callback- dialstring	Sets the telephone number for a callback (for example: callback-dialstring= 408-555-1212). Value is NULL, or a dial-string. A NULL value indicates that the service might choose to get the dial string through other means. Used with service=arap, service=slip, service=ppp, service=shell. Not valid for ISDN.	no	yes	yes	yes	yes	yes	yes
callback-line	The number of a TTY line to use for callback (for example: callback-line=4). Used with service=arap, service=slip, service=ppp, service=shell. Not valid for ISDN.	no	yes	yes	yes	yes	yes	yes
callback-rotary	The number of a rotary group (between 0 and 100 inclusive) to use for callback (for example: callback-rotary=34). Used with service=arap, service=slip, service=ppp, service=shell. Not valid for ISDN.	no	yes	yes	yes	yes	yes	yes
cmd-arg=x	An argument to a shell (EXEC) command. This indicates an argument for the shell command that is to be run. Multiple cmd-arg attributes can be specified, and they are order dependent. Note This TACACS+ AV pair cannot be used with RADIUS attribute 26.	yes	yes	yes	yes	yes	yes	yes

Attribute	Description	11.0	11.1	11.2	11.3	12.0	12.1	12.2
cmd=x	A shell (EXEC) command. This indicates the command name for a shell command that is to be run. This attribute must be specified if service equals "shell." A NULL value indicates that the shell itself is being referred to. Note This TACACS+ AV pair cannot be used with RADIUS attribute 26.	yes	yes	yes	yes	yes	yes	yes
data-service	Used with the service=outbound and protocol=ip.	no	no	no	no	no	yes	yes
dial-number	Defines the number to dial. Used with the service=outbound and protocol=ip.	no	no	no	no	no	yes	yes
dns-servers=	Identifies a DNS server (primary or secondary) that can be requested by Microsoft PPP clients from the network access server during IPCP negotiation. To be used with service=ppp and protocol=ip. The IP address identifying each DNS server is entered in dotted decimal format.	no	no	no	yes	yes	yes	yes
force-56	Determines whether the network access server uses only the 56 K portion of a channel, even when all 64 K appear to be available. To turn on this attribute, use the "true" value (force-56=true). Any other value is treated as false. Used with the service=outbound and protocol=ip.	no	no	no	no	no	yes	yes
gw-password	Specifies the password for the home gateway during the L2F tunnel authentication. Used with service=ppp and protocol=vpdn.	no	no	yes	yes	yes	yes	yes
idletime=x	Sets a value, in minutes, after which an idle session is terminated. A value of zero indicates no timeout.	no	yes	yes	yes	yes	yes	yes
inacl#<n>	ASCII access list identifier for an input access list to be installed and applied to an interface for the duration of the current connection. Used with service=ppp and protocol=ip, and service service=ppp and protocol=ipx. Per-user access lists do not currently work with ISDN interfaces.	no	no	no	yes	yes	yes	yes
inacl=x	ASCII identifier for an interface input access list. Used with service=ppp and protocol=ip. Per-user access lists do not currently work with ISDN interfaces.	yes	yes	yes	yes	yes	yes	yes
interface-config#<n>	Specifies user-specific AAA interface configuration information with Virtual Profiles. The information that follows the equal sign (=) can be any Cisco IOS interface configuration command. Multiple instances of the attributes are allowed, but each instance must have a unique number. Used with service=ppp and protocol=lcp. Note This attribute replaces the "interface-config=" attribute.	no	no	no	yes	yes	yes	yes
ip-addresses	Space-separated list of possible IP addresses that can be used for the end-point of a tunnel. Used with service=ppp and protocol=vpdn.	no	no	yes	yes	yes	yes	yes

Attribute	Description	11.0	11.1	11.2	11.3	12.0	12.1	12.2
l2tp-busy-disconnect	If a vpdn-group on an LNS uses a virtual-template that is configured to be pre-cloned, this attribute will control the disposition of a new L2TP session that finds no pre-cloned interface to which to connect. If the attribute is true (the default), the session will be disconnected by the LNS. Otherwise, a new interface will be cloned from the virtual-template. Used with service=ppp and protocol=vpdn.	no	no	no	no	no	yes	yes
l2tp-cm-local-window-size	Specifies the maximum receive window size for L2TP control messages. This value is advertised to the peer during tunnel establishment. Used with service=ppp and protocol=vpdn.	no	no	no	no	no	yes	yes
l2tp-drop-out-of-order	Respects sequence numbers on data packets by dropping those that are received out of order. This does not ensure that sequence numbers will be sent on data packets, just how to handle them if they are received. Used with service=ppp and protocol=vpdn.	no	no	no	no	no	yes	yes
l2tp-hello- interval	Specifies the number of seconds for the hello keepalive interval. Hello packets are sent when no data has been sent on a tunnel for the number of seconds configured here. Used with service=ppp and protocol=vpdn.	no	no	no	no	no	yes	yes
l2tp-hidden-avp	When enabled, sensitive AVPs in L2TP control messages are scrambled or hidden. Used with service=ppp and protocol=vpdn.	no	no	no	no	no	yes	yes
l2tp-nosession-timeout	Specifies the number of seconds that a tunnel will stay active with no sessions before timing out and shutting down. Used with service=ppp and protocol=vpdn.	no	no	no	no	no	yes	yes
l2tp-tos-reflect	Copies the IP ToS field from the IP header of each payload packet to the IP header of the tunnel packet for packets entering the tunnel at the LNS. Used with service=ppp and protocol=vpdn.	no	no	no	no	no	yes	yes
l2tp-tunnel- authen	If this attribute is set, it performs L2TP tunnel authentication. Used with service=ppp and protocol=vpdn.	no	no	no	no	no	yes	yes
l2tp-tunnel-password	Shared secret used for L2TP tunnel authentication and AVP hiding. Used with service=ppp and protocol=vpdn.	no	no	no	no	no	yes	yes
l2tp-udp- checksum	This is an authorization attribute and defines whether L2TP should perform UDP checksums for data packets. Valid values are “yes” and “no.” The default is no. Used with service=ppp and protocol=vpdn.	no	no	no	no	no	yes	yes

Attribute	Description	11.0	11.1	11.2	11.3	12.0	12.1	12.2
link- compression=	Defines whether to turn on or turn off “stac” compression over a PPP link. Used with service=ppp. Link compression is defined as a numeric value as follows: <ul style="list-style-type: none"> • 0: None • 1: Stac • 2: Stac-Draft-9 • 3: MS-Stac 	no	no	no	yes	yes	yes	yes
load-threshold=<n>	Sets the load threshold for the caller at which additional links are either added to or deleted from the multilink bundle. If the load goes above the specified value, additional links are added. If the load goes below the specified value, links are deleted. Used with service=ppp and protocol=multilink. The range for <n> is from 1 to 255.	no	no	no	yes	yes	yes	yes
map-class	Allows the user profile to reference information configured in a map class of the same name on the network access server that dials out. Used with the service=outbound and protocol=ip.	no	no	no	no	no	yes	yes
max-links=<n>	Restricts the number of links that a user can have in a multilink bundle. Used with service=ppp and protocol=multilink. The range for <n> is from 1 to 255.	no	no	no	yes	yes	yes	yes
min-links	Sets the minimum number of links for MLP. Used with service=ppp and protocol=multilink, protocol=vpdn.	no	no	no	no	no	yes	yes
nas-password	Specifies the password for the network access server during the L2F tunnel authentication. Used with service=ppp and protocol=vpdn.	no	no	yes	yes	yes	yes	yes
nocallback-verify	Indicates that no callback verification is required. The only valid value for this parameter is 1 (for example, nocallback-verify=1). Used with service=arap, service=slip, service=ppp, service=shell. There is no authentication on callback. Not valid for ISDN.	no	yes	yes	yes	yes	yes	yes
noescape=x	Prevents user from using an escape character. Used with service=shell. Can be either true or false (for example, noescape=true).	yes	yes	yes	yes	yes	yes	yes
nohangup=x	Used with service=shell. Specifies the nohangup option, which means that after an EXEC shell is terminated, the user is presented with another login (username) prompt. Can be either true or false (for example, nohangup=false).	yes	yes	yes	yes	yes	yes	yes
old-prompts	Allows providers to make the prompts in TACACS+ appear identical to those of earlier systems (TACACS and Extended TACACS). This allows administrators to upgrade from TACACS or Extended TACACS to TACACS+ transparently to users.	yes	yes	yes	yes	yes	yes	yes

Attribute	Description	11.0	11.1	11.2	11.3	12.0	12.1	12.2
outacl#<n>	ASCII access list identifier for an interface output access list to be installed and applied to an interface for the duration of the current condition. Used with service=ppp and protocol=ip, and service service=ppp and protocol=ipx. Per-user access lists do not currently work with ISDN interfaces.	no	no	no	yes	yes	yes	yes
outacl=x	ASCII identifier for an interface output access list. Used with service=ppp and protocol=ip, and service service=ppp and protocol=ipx. Contains an IP output access list for SLIP or PPP/IP (for example, outacl=4). The access list itself must be preconfigured on the router. Per-user access lists do not currently work with ISDN interfaces.	yes (PPP/IP only)	yes	yes	yes	yes	yes	yes
pool-def#<n>	Defines IP address pools on the network access server. Used with service=ppp and protocol=ip.	no	no	no	yes	yes	yes	yes
pool-timeout=	Defines (in conjunction with pool-def) IP address pools on the network access server. During IPCP address negotiation, if an IP pool name is specified for a user (see the addr-pool attribute), a check is made to see if the named pool is defined on the network access server. If it is, the pool is consulted for an IP address. Used with service=ppp and protocol=ip.	no	no	yes	yes	yes	yes	yes
port-type	Indicates the type of physical port the network access server is using to authenticate the user. Physical ports are indicated by a numeric value as follows: <ul style="list-style-type: none"> • 0: Asynchronous • 1: Synchronous • 2: ISDN-Synchronous • 3: ISDN-Asynchronous (V.120) • 4: ISDN- Asynchronous (V.110) • 5: Virtual Used with service=any and protocol=aaa.	no	no	no	no	no	yes	yes
ppp-vj-slot-compression	Instructs the Cisco router not to use slot compression when sending VJ-compressed packets over a PPP link.	no	no	no	yes	yes	yes	yes
priv-lvl=x	Privilege level to be assigned for the EXEC. Used with service=shell. Privilege levels range from 0 to 15, with 15 being the highest.	yes	yes	yes	yes	yes	yes	yes
protocol=x	A protocol that is a subset of a service. An example would be any PPP NCP. Currently known values are lcp , ip , ipx , atalk , vines , lat , xremote , tn3270 , telnet , rlogin , pad , vpdn , osicp , deccp , ccp , cdp , bridging , xns , nbf , bap , multilink , and unknown .	yes	yes	yes	yes	yes	yes	yes

Attribute	Description	11.0	11.1	11.2	11.3	12.0	12.1	12.2
proxyacl#<n>	Allows users to configure the downloadable user profiles (dynamic ACLs) by using the authentication proxy feature so that users can have the configured authorization to permit traffic going through the configured interfaces. Used with the service=shell and protocol=exec.	no	no	no	no	no	yes	yes
route	Specifies a route to be applied to an interface. Used with service=slip, service=ppp, and protocol=ip. During network authorization, the route attribute can be used to specify a per-user static route, to be installed by TACACS+ as follows: route="dst_address mask [gateway]" This indicates a temporary static route that is to be applied. The <i>dst_address</i> , <i>mask</i> , and <i>gateway</i> are expected to be in the usual dotted-decimal notation, with the same meanings as in the familiar ip route configuration command on a network access server. If <i>gateway</i> is omitted, the peer's address is the gateway. The route is expunged when the connection terminates.	no	yes	yes	yes	yes	yes	yes
route#<n>	Like the route AV pair, this specifies a route to be applied to an interface, but these routes are numbered, allowing multiple routes to be applied. Used with service=ppp and protocol=ip, and service=ppp and protocol=ipx.	no	no	no	yes	yes	yes	yes
routing=x	Specifies whether routing information is to be propagated to and accepted from this interface. Used with service=slip, service=ppp, and protocol=ip. Equivalent in function to the /routing flag in SLIP and PPP commands. Can either be true or false (for example, routing=true).	yes	yes	yes	yes	yes	yes	yes
rte-fltr-in#<n>	Specifies an input access list definition to be installed and applied to routing updates on the current interface for the duration of the current connection. Used with service=ppp and protocol=ip, and with service=ppp and protocol=ipx.	no	no	no	yes	yes	yes	yes
rte-fltr-out#<n>	Specifies an output access list definition to be installed and applied to routing updates on the current interface for the duration of the current connection. Used with service=ppp and protocol=ip, and with service=ppp and protocol=ipx.	no	no	no	yes	yes	yes	yes
sap#<n>	Specifies static Service Advertising Protocol (SAP) entries to be installed for the duration of a connection. Used with service=ppp and protocol=ipx.	no	no	no	yes	yes	yes	yes
sap-fltr-in#<n>	Specifies an input SAP filter access list definition to be installed and applied on the current interface for the duration of the current connection. Used with service=ppp and protocol=ipx.	no	no	no	yes	yes	yes	yes

Attribute	Description	11.0	11.1	11.2	11.3	12.0	12.1	12.2
sap-fltr-out#<n>	Specifies an output SAP filter access list definition to be installed and applied on the current interface for the duration of the current connection. Used with service=ppp and protocol=ipx.	no	no	no	yes	yes	yes	yes
send-auth	Defines the protocol to use (PAP or CHAP) for username-password authentication following CLID authentication. Used with service=any and protocol=aaa.	no	no	no	no	no	yes	yes
send-secret	Specifies the password that the NAS needs to respond to a chap/pap request from the remote end of a connection on an outgoing call. Used with service=ppp and protocol=ip.	no	no	no	no	no	yes	yes
service=x	The primary service. Specifying a service attribute indicates that this is a request for authorization or accounting of that service. Current values are slip , ppp , arap , shell , tty-daemon , connection , and system . This attribute must always be included.	yes	yes	yes	yes	yes	yes	yes
source-ip=x	Used as the source IP address of all VPDN packets generated as part of a VPDN tunnel. This is equivalent to the Cisco vpdn outgoing global configuration command.	no	no	yes	yes	yes	yes	yes
spi	Carries the authentication information needed by the home agent to authenticate a mobile node during registration. The information is in the same syntax as the ip mobile secure host <addr> configuration command. Basically it contains the rest of the configuration command that follows that string, verbatim. It provides the Security Parameter Index (SPI), key, authentication algorithm, authentication mode, and replay protection timestamp range. Used with the service=mobileip and protocol=ip.	no	no	no	no	no	yes	yes
timeout=x	The number of minutes before an EXEC or ARA session disconnects (for example, timeout=60). A value of zero indicates no timeout. Used with service=arap.	yes	yes	yes	yes	yes	yes	yes
tunnel-id	Specifies the username that will be used to authenticate the tunnel over which the individual user MID will be projected. This is analogous to the <i>remote name</i> in the vpdn outgoing command. Used with service=ppp and protocol=vpdn.	no	no	yes	yes	yes	yes	yes
wins-servers=	Identifies a Windows NT server that can be requested by Microsoft PPP clients from the network access server during IPCP negotiation. To be used with service=ppp and protocol=ip. The IP address identifying each Windows NT server is entered in dotted decimal format.	no	no	no	yes	yes	yes	yes
zonelist=x	A numeric zonelist value. Used with service=arap. Specifies an AppleTalk zonelist for ARA (for example, zonelist=5).	yes	yes	yes	yes	yes	yes	yes

See Configuring TACACS+ module for the documents used to configure TACACS+, and TACACS+ authentication and authorization.

TACACS Accounting AV Pairs

The following table lists and describes the supported TACACS+ accounting AV pairs and specifies the Cisco IOS release in which they are implemented.

Table 107: Supported TACACS+ Accounting AV Pairs

Attribute	Description	11.0	11.1	11.2	11.3	12.0	12.1	12.2
Abort-Cause	If the fax session is terminated, indicates the system component that signaled the termination. Examples of system components that could trigger a termination are FAP (Fax Application Process), TIFF (the TIFF reader or the TIFF writer), fax-mail client, fax-mail server, ESMTP client, or ESMTP server.	no	no	no	no	no	yes	yes
bytes_in	The number of input bytes transferred during this connection.	yes	yes	yes	yes	yes	yes	yes
bytes_out	The number of output bytes transferred during this connection.	yes	yes	yes	yes	yes	yes	yes
Call-Type	Describes the type of fax activity: fax receive or fax send.	no	no	no	no	no	yes	yes
cmd	The command the user executed.	yes	yes	yes	yes	yes	yes	yes
data-rate	This AV pair has been renamed. See nas-rx-speed.							
disc-cause	Specifies the reason a connection was taken off-line. The Disconnect-Cause attribute is sent in accounting-stop records. This attribute also causes stop records to be generated without first generating start records if disconnection occurs before authentication is performed. Refer to the following table (Disconnect Cause Extensions) for a list of Disconnect-Cause values and their meanings.	no	no	no	yes	yes	yes	yes
disc-cause-ext	Extends the disc-cause attribute to support vendor-specific reasons why a connection was taken off-line.	no	no	no	yes	yes	yes	yes
elapsed_time	The elapsed time in seconds for the action. Useful when the device does not keep real time.	yes	yes	yes	yes	yes	yes	yes
Email-Server-Address	Indicates the IP address of the e-mail server handling the on-ramp fax-mail message.	no	no	no	no	no	yes	yes
Email-Server-Ack-Flag	Indicates that the on-ramp gateway has received a positive acknowledgment from the e-mail server accepting the fax-mail message.	no	no	no	no	no	yes	yes
event	Information included in the accounting packet that describes a state change in the router. Events described are accounting starting and accounting stopping.	yes	yes	yes	yes	yes	yes	yes
Fax-Account-Id-Origin	Indicates the account ID origin as defined by system administrator for the mmp aa receive-id or the mmp aa send-id command.	no	no	no	no	no	yes	yes
Fax-Auth-Status	Indicates whether or not authentication for this fax session was successful. Possible values for this field are success, failed, bypassed, or unknown.	no	no	no	no	no	yes	yes

Attribute	Description	11.0	11.1	11.2	11.3	12.0	12.1	12.2
Fax-Connect-Speed	Indicates the modem speed at which this fax-mail was initially transmitted or received. Possible values are 1200, 4800, 9600, and 14400.	no	no	no	no	no	yes	yes
Fax-Coverpage-Flag	Indicates whether or not a cover page was generated by the off-ramp gateway for this fax session. True indicates that a cover page was generated; false means that a cover page was not generated.	no	no	no	no	no	yes	yes
Fax-Dsn-Address	Indicates the address to which DSNs will be sent.	no	no	no	no	no	yes	yes
Fax-Dsn-Flag	Indicates whether or not DSN has been enabled. True indicates that DSN has been enabled; false means that DSN has not been enabled.	no	no	no	no	no	yes	yes
Fax-Mdn-Address	Indicates the address to which MDNs will be sent.	no	no	no	no	no	yes	yes
Fax-Mdn-Flag	Indicates whether or not message delivery notification (MDN) has been enabled. True indicates that MDN had been enabled; false means that MDN had not been enabled.	no	no	no	no	no	yes	yes
Fax-Modem-Time	Indicates the amount of time in seconds the modem sent fax data (x) and the amount of time in seconds of the total fax session (y), which includes both fax-mail and PSTN time, in the form x/y. For example, 10/15 means that the transfer time took 10 seconds, and the total fax session took 15 seconds.	no	no	no	no	no	yes	yes
Fax-Msg-Id=	Indicates a unique fax message identification number assigned by Store and Forward Fax.	no	no	no	no	no	yes	yes
Fax-Pages	Indicates the number of pages transmitted or received during this fax session. This page count includes cover pages.	no	no	no	no	no	yes	yes
Fax-Process-Abort-Flag	Indicates that the fax session was terminated or successful. True means that the session was terminated; false means that the session was successful.	no	no	no	no	no	yes	yes
Fax-Recipient-Count	Indicates the number of recipients for this fax transmission. Until e-mail servers support Session mode, the number should be 1.	no	no	no	no	no	yes	yes
Gateway-Id	Indicates the name of the gateway that processed the fax session. The name appears in the following format: hostname.domain-name	no	no	no	no	no	yes	yes
mlp-links-max	Gives the count of links which are known to have been in a given multilink session at the time the accounting record is generated.	no	no	no	yes	yes	yes	yes
mlp-sess-id	Reports the identification number of the multilink bundle when the session closes. This attribute applies to sessions that are part of a multilink bundle. This attribute is sent in authentication-response packets.	no	no	no	yes	yes	yes	yes
nas-rx-speed	Specifies the average number of bits per second over the course of the connection's lifetime. This attribute is sent in accounting-stop records.	no	no	no	yes	yes	yes	yes
nas-tx-speed	Reports the transmit speed negotiated by the two modems.	no	no	no	yes	yes	yes	yes
paks_in	The number of input packets transferred during this connection.	yes	yes	yes	yes	yes	yes	yes

Attribute	Description	11.0	11.1	11.2	11.3	12.0	12.1	12.2
paks_out	The number of output packets transferred during this connection.	yes	yes	yes	yes	yes	yes	yes
port	The port the user was logged in to.	yes	yes	yes	yes	yes	yes	yes
Port-Used	Indicates the slot/port number of the Cisco AS5300 used to either transmit or receive this fax-mail.	no	no	no	no	no	yes	yes
pre-bytes-in	Records the number of input bytes before authentication. This attribute is sent in accounting-stop records.	no	no	no	yes	yes	yes	yes
pre-bytes-out	Records the number of output bytes before authentication. This attribute is sent in accounting-stop records.	no	no	no	yes	yes	yes	yes
pre-paks-in	Records the number of input packets before authentication. This attribute is sent in accounting-stop records.	no	no	no	yes	yes	yes	yes
pre-paks-out	Records the number of output packets before authentication. The Pre-Output-Packets attribute is sent in accounting-stop records.	no	no	no	yes	yes	yes	yes
pre-session-time	Specifies the length of time, in seconds, from when a call first connects to when it completes authentication.	no	no	no	yes	yes	yes	yes
priv_level	The privilege level associated with the action.	yes	yes	yes	yes	yes	yes	yes
protocol	The protocol associated with the action.	yes	yes	yes	yes	yes	yes	yes
reason	Information included in the accounting packet that describes the event that caused a system change. Events described are system reload, system shutdown, or when accounting is reconfigured (turned on or off).	yes	yes	yes	yes	yes	yes	yes
service	The service the user used.	yes	yes	yes	yes	yes	yes	yes
start_time	The time the action started (in seconds since the epoch, 12:00 a.m. Jan 1 1970). The clock must be configured to receive this information.	yes	yes	yes	yes	yes	yes	yes
stop_time	The time the action stopped (in seconds since the epoch.) The clock must be configured to receive this information.	yes	yes	yes	yes	yes	yes	yes
task_id	Start and stop records for the same event must have matching (unique) task_id numbers.	yes	yes	yes	yes	yes	yes	yes
timezone	The time zone abbreviation for all timestamps included in this packet.	yes	yes	yes	yes	yes	yes	yes
xmit-rate	This AV pair has been renamed. See nas-tx-speed.							

The following table lists the cause codes and descriptions for the Disconnect Cause Extended (disc-cause-ext) attribute.

Table 108: Disconnect Cause Extensions

Cause Codes	Description	11.0	11.1	11.2	11.3	12.0	12.1	12.2	12.3
1000 - No Reason	No reason for the disconnect.	no	no	no	no	yes	yes	yes	yes
1001 - No Disconnect	The event was not a disconnect.	no	no	no	no	yes	yes	yes	yes
1002 - Unknown	The reason for the disconnect is unknown. This code can appear when the remote connection goes down.	no	no	no	no	yes	yes	yes	yes
1003 - Call Disconnect	The call has disconnected.	no	no	no	no	yes	yes	yes	yes
1004 - CLID Auth Fail	Calling line ID (CLID) authentication has failed.	no	no	no	no	yes	yes	yes	yes
1009 - No Modem Available	The modem is not available.	no	no	no	no	yes	yes	yes	yes
1010 - No Carrier	The modem never detected data carrier detect (DCD). This code can appear if a disconnect occurs during the initial modem connection.	no	no	no	no	yes	yes	yes	yes
1011 - Lost Carrier	The modem detected DCD but became inactive. This code can appear if a disconnect occurs during the initial modem connection.	no	no	no	no	yes	yes	yes	yes
1012 - No Modem Results	The result codes could not be parsed. This code can appear if a disconnect occurs during the initial modem connection.	no	no	no	no	yes	yes	yes	yes
1020 - TS User Exit	The user exited normally from the terminal server. This code is related to immediate Telnet and raw TCP disconnects during a terminal server session.	no	no	no	no	yes	yes	yes	yes
1021 - Idle Timeout	The user exited from the terminal server because the idle timer expired. This code is related to immediate Telnet and raw TCP disconnects during a terminal server session.	no	no	no	no	yes	yes	yes	yes
1022 - TS Exit Telnet	The user exited normally from a Telnet session. This code is related to immediate Telnet and raw TCP disconnects during a terminal server session.	no	no	no	no	yes	yes	yes	yes
1023 - TS No IP Addr	The user could not switch to Serial Line Internet Protocol (SLIP) or PPP because the remote host had no IP address or because the dynamic pool could not assign one. This code is related to immediate Telnet and raw TCP disconnects during a terminal server session.	no	no	no	no	yes	yes	yes	yes
1024 - TS TCP Raw Exit	The user exited normally from a raw TCP session. This code is related to immediate Telnet and raw TCP disconnects during a terminal server session.	no	no	no	no	yes	yes	yes	yes

Cause Codes	Description	11.0	11.1	11.2	11.3	12.0	12.1	12.2	12.3
1025 - TS Bad Password	The login process ended because the user failed to enter a correct password after three attempts. This code is related to immediate Telnet and raw TCP disconnects during a terminal server session.	no	no	no	no	yes	yes	yes	yes
1026 - TS No TCP Raw	The raw TCP option is not enabled. This code is related to immediate Telnet and raw TCP disconnects during a terminal server session.	no	no	no	no	yes	yes	yes	yes
1027 - TS CNTL-C	The login process ended because the user typed Ctrl-C. This code is related to immediate Telnet and raw TCP disconnects during a terminal server session.	no	no	no	no	yes	yes	yes	yes
1028 - TS Session End	The terminal server session has ended. This code is related to immediate Telnet and raw TCP disconnects during a terminal server session.	no	no	no	no	yes	yes	yes	yes
1029 - TS Close Vconn	The user closed the virtual connection. This code is related to immediate Telnet and raw TCP disconnects during a terminal server session.	no	no	no	no	yes	yes	yes	yes
1030 - TS End Vconn	The virtual connection has ended. This code is related to immediate Telnet and raw TCP disconnects during a terminal server session.	no	no	no	no	yes	yes	yes	yes
1031 - TS Rlogin Exit	The user exited normally from an Rlogin session. This code is related to immediate Telnet and raw TCP disconnects during a terminal server session.	no	no	no	no	yes	yes	yes	yes
1032 - TS Rlogin Opt Invalid	The user selected an invalid Rlogin option. This code is related to immediate Telnet and raw TCP disconnects during a terminal server session.	no	no	no	no	yes	yes	yes	yes
1033 - TS Insuff Resources	The access server has insufficient resources for the terminal server session. This code is related to immediate Telnet and raw TCP disconnects during a terminal server session.	no	no	no	no	yes	yes	yes	yes
1040 - PPP LCP Timeout	PPP link control protocol (LCP) negotiation timed out while waiting for a response from a peer. This code concerns PPP connections.	no	no	no	no	yes	yes	yes	yes
1041 - PPP LCP Fail	There was a failure to converge on PPP LCP negotiations. This code concerns PPP connections.	no	no	no	no	yes	yes	yes	yes
1042 - PPP Pap Fail	PPP Password Authentication Protocol (PAP) authentication failed. This code concerns PPP connections.	no	no	no	no	yes	yes	yes	yes
1043 - PPP CHAP Fail	PPP Challenge Handshake Authentication Protocol (CHAP) authentication failed. This code concerns PPP connections.	no	no	no	no	yes	yes	yes	yes
1044 - PPP Remote Fail	Authentication failed from the remote server. This code concerns PPP sessions.	no	no	no	no	yes	yes	yes	yes

Cause Codes	Description	11.0	11.1	11.2	11.3	12.0	12.1	12.2	12.3
1045 - PPP Receive Term	The peer sent a PPP termination request. This code concerns PPP connections.	no	no	no	no	yes	yes	yes	yes
PPP LCP Close (1046)	LCP got a close request from the upper layer while LCP was in an open state. This code concerns PPP connections.	no	no	no	no	yes	yes	yes	yes
1047 - PPP No NCP	LCP closed because no NCPs were open. This code concerns PPP connections.	no	no	no	no	yes	yes	yes	yes
1048 - PPP MP Error	LCP closed because it could not determine to which Multilink PPP bundle that it should add the user. This code concerns PPP connections.	no	no	no	no	yes	yes	yes	yes
1049 - PPP Max Channels	LCP closed because the access server could not add any more channels to an MP session. This code concerns PPP connections.	no	no	no	no	yes	yes	yes	yes
1050 - TS Tables Full	The raw TCP or Telnet internal session tables are full. This code relates to immediate Telnet and raw TCP disconnects and contains more specific information than the Telnet and TCP codes listed earlier in this table.	no	no	no	no	yes	yes	yes	yes
1051 - TS Resource Full	Internal resources are full. This code relates to immediate Telnet and raw TCP disconnects and contains more specific information than the Telnet and TCP codes listed earlier in this table.	no	no	no	no	yes	yes	yes	yes
1052 - TS Invalid IP Addr	The IP address for the Telnet host is invalid. This code relates to immediate Telnet and raw TCP disconnects and contains more specific information than the Telnet and TCP codes listed earlier in this table.	no	no	no	no	yes	yes	yes	yes
1053 - TS Bad Hostname	The access server could not resolve the host name. This code relates to immediate Telnet and raw TCP disconnects and contains more specific information than the Telnet and TCP codes listed earlier in this table.	no	no	no	no	yes	yes	yes	yes
1054 - TS Bad Port	The access server detected a bad or missing port number. This code relates to immediate Telnet and raw TCP disconnects and contains more specific information than the Telnet and TCP codes listed earlier in this table.	no	no	no	no	yes	yes	yes	yes
1060 - TCP Reset	The host reset the TCP connection. The TCP stack can return this disconnect code during an immediate Telnet or raw TCP session.	no	no	no	no	yes	yes	yes	yes
1061 - TCP Connection Refused	The host refused the TCP connection. The TCP stack can return this disconnect code during an immediate Telnet or raw TCP session.	no	no	no	no	yes	yes	yes	yes

Cause Codes	Description	11.0	11.1	11.2	11.3	12.0	12.1	12.2	12.3
1062 - TCP Timeout	The TCP connection timed out. The TCP stack can return this disconnect code during an immediate Telnet or raw TCP session.	no	no	no	no	yes	yes	yes	yes
1063 - TCP Foreign Host Close	A foreign host closed the TCP connection. The TCP stack can return this disconnect code during an immediate Telnet or raw TCP session.	no	no	no	no	yes	yes	yes	yes
1064 - TCP Net Unreachable	The TCP network was unreachable. The TCP stack can return this disconnect code during an immediate Telnet or raw TCP session.	no	no	no	no	yes	yes	yes	yes
1065 - TCP Host Unreachable	The TCP host was unreachable. The TCP stack can return this disconnect code during an immediate Telnet or raw TCP session.	no	no	no	no	yes	yes	yes	yes
1066 - TCP Net Admin Unreachable	The TCP network was administratively unreachable. The TCP stack can return this disconnect code during an immediate Telnet or raw TCP session.	no	no	no	no	yes	yes	yes	yes
1067 - TCP Host Admin Unreachable	The TCP host was administratively unreachable. The TCP stack can return this disconnect code during an immediate Telnet or raw TCP session.	no	no	no	no	yes	yes	yes	yes
1068 - TCP Port Unreachable	The TCP port was unreachable. The TCP stack can return this disconnect code during an immediate Telnet or raw TCP session.	no	no	no	no	yes	yes	yes	yes
1100 - Session Timeout	The session timed out because there was no activity on a PPP link. This code applies to all session types.	no	no	no	no	yes	yes	yes	yes
1101 - Security Fail	The session failed for security reasons. This code applies to all session types.	no	no	no	no	yes	yes	yes	yes
1102 - Callback	The session ended for callback. This code applies to all session types.	no	no	no	no	yes	yes	yes	yes
1120 - Unsupported	One end refused the call because the protocol was disabled or unsupported. This code applies to all session types.	no	no	no	no	yes	yes	yes	yes
1150 - Radius Disc	The RADIUS server requested the disconnect.	no	no	no	no	yes	yes	yes	yes
1151 - Local Admin Disc	The local administrator has disconnected.	no	no	no	no	yes	yes	yes	yes
1152 - SNMP Disc	Simple Network Management Protocol (SNMP) has disconnected.	no	no	no	no	yes	yes	yes	yes
1160 - V110 Retries	The allowed retries for V110 synchronization have been exceeded.	no	no	no	no	yes	yes	yes	yes
1170 - PPP Auth Timeout	Authentication timeout. This code applies to PPP sessions.	no	no	no	no	yes	yes	yes	yes
1180 - Local Hangup	The call disconnected as the result of a local hangup.	no	no	no	no	yes	yes	yes	yes

Cause Codes	Description	11.0	11.1	11.2	11.3	12.0	12.1	12.2	12.3
1185 - Remote Hangup	The call disconnected because the remote end hung up.	no	no	no	no	yes	yes	yes	yes
1190 - T1 Quiesced	The call disconnected because the T1 line that carried it was quiesced.	no	no	no	no	yes	yes	yes	yes
1195 - Call Duration	The call disconnected because the call duration exceeded the maximum amount of time allowed by the Max Call Mins or Max DS0 Mins parameter on the access server.	no	no	no	no	yes	yes	yes	yes
1600 - VPDN User Disconnect	The user disconnected. This value applies to virtual private dial-up network (VPDN) sessions.	no	no	no	no	no	no	yes	yes
1601 - VPDN Carrier Loss	Carrier loss has occurred. This code applies to VPDN sessions.	no	no	no	no	no	no	yes	yes
1602 - VPDN No Resources	There are no resources. This code applies to VPDN sessions.	no	no	no	no	no	no	yes	yes
1603 - VPDN Bad Control Packet	The control packet is invalid. This code applies to VPDN sessions.	no	no	no	no	no	no	yes	yes
1604 - VPDN Admin Disconnect	The administrator disconnected. This code applies to VPDN sessions.	no	no	no	no	no	no	yes	yes
1605 - VPDN Tunnel Down/Setup Fail	The tunnel is down or the setup failed. This code applies to VPDN sessions.	no	no	no	no	no	no	yes	yes
1606 - VPDN Local PPP Disconnect	There was a local PPP disconnect. This code applies to VPDN sessions.	no	no	no	no	no	no	yes	yes
1607 - VPDN Softshut/Session Limit	New sessions cannot be established on the VPN tunnel. This code applies to VPDN sessions.	no	no	no	no	no	no	yes	yes
1608 - VPDN Call Redirected	The call was redirected. This code applies to VPDN sessions.	no	no	no	no	no	no	yes	yes
1801 - Q850 Unassigned Number	The number has not been assigned. This code applies to ISDN or modem calls that came in over ISDN.	no	no	no	no	no	no	no	yes
1802 - Q850 No Route	The equipment that is sending this code has received a request to route the call through a particular transit network that it does not recognize. The equipment that is sending this code does not recognize the transit network because either the transit network does not exist or because that particular transit network, while it does exist, does not serve the equipment that is sending this code. This code applies to ISDN or modem calls that came in over ISDN.	no	no	no	no	no	no	no	yes
1803 - Q850 No Route To Destination	The called party cannot be reached because the network through which the call has been routed does not serve the destination that is desired. This code applies to ISDN or modem calls that came in over ISDN.	no	no	no	no	no	no	no	yes

Cause Codes	Description	11.0	11.1	11.2	11.3	12.0	12.1	12.2	12.3
1806 - Q850 Channel Unacceptable	The channel that has been most recently identified is not acceptable to the sending entity for use in this call. This code applies to ISDN or modem calls that came in over ISDN.	no	no	no	no	no	no	no	yes
1816 - Q850 Normal Clearing	The call is being cleared because one of the users who is involved in the call has requested that the call be cleared. This code applies to ISDN or modem calls that came in over ISDN.	no	no	no	no	no	no	no	yes
1817 - Q850 User Busy	The called party is unable to accept another call because the user-busy condition has been encountered. This code may be generated by the called user or by the network. In the case of the user, the user equipment is compatible with the call. This code applies to ISDN or modem calls that came in over ISDN.	no	no	no	no	no	no	no	yes
1818 - Q850 No User Responding	Used when a called party does not respond to a call-establishment message with either an alerting or connect indication within the prescribed period of time that was allocated. This code applies to ISDN or modem calls that came in over ISDN.	no	no	no	no	no	no	no	yes
1819 - Q850 No User Answer	The called party has been alerted but does not respond with a connect indication within a prescribed period of time. This code applies to ISDN or modem calls that came in over ISDN.	no	no	no	no	no	no	no	yes
1821 - Q850 Call Rejected	The equipment that is sending this code does not wish to accept this call although it could have accepted the call because the equipment that is sending this code is neither busy nor incompatible. This code may also be generated by the network, indicating that the call was cleared due to a supplementary service constraint. The diagnostic field may contain additional information about the supplementary service and reason for rejection. This code applies to ISDN or modem calls that came in over ISDN.	no	no	no	no	no	no	no	yes
1822 - Q850 Number Changed	The number that is indicated for the called party is no longer assigned. The new called party number may optionally be included in the diagnostic field. This code applies to ISDN or modem calls that came in over ISDN.	no	no	no	no	no	no	no	yes
1827 - Q850 Destination Out of Order	The destination that was indicated by the user cannot be reached because the interface to the destination is not functioning correctly. The term "not functioning correctly" indicates that a signaling message was unable to be delivered to the remote party. This code applies to ISDN or modem calls that came in over ISDN.	no	no	no	no	no	no	no	yes
1828 - Q850 Invalid Number Format	The called party cannot be reached because the called party number is not in a valid format or is not complete. This code applies to ISDN or modem calls that came in over ISDN.	no	no	no	no	no	no	no	yes

Cause Codes	Description	11.0	11.1	11.2	11.3	12.0	12.1	12.2	12.3
1829 - Q850 Facility Rejected	This code is returned when a supplementary service that was requested by the user cannot be provided by the network. This code applies to ISDN or modem calls that have come in over ISDN.	no	no	no	no	no	no	no	yes
1830 - Q850 Responding to Status Enquiry	This code is included in the STATUS message when the reason for generating the STATUS message was the prior receipt of a STATUS ENQUIRY message. This code applies to ISDN or modem calls that came in over ISDN.	no	no	no	no	no	no	no	yes
1831 - Q850 Unspecified Cause	No other code applies. This code applies to ISDN or modem calls that came in over ISDN.	no	no	no	no	no	no	no	yes
1834 - Q850 No Circuit Available	No circuit or channel is available to handle the call. This code applies to ISDN or modem calls that came in over ISDN.	no	no	no	no	no	no	no	yes
1838 - Q850 Network Out of Order	The network is not functioning correctly and the condition is likely to last a relatively long period of time. This code applies to ISDN or modem calls that came in over ISDN.	no	no	no	no	no	no	no	yes
1841 - Q850 Temporary Failure	The network is not functioning correctly and the condition is not likely to last a long period of time. This code applies to ISDN or modem calls that came in over ISDN.	no	no	no	no	no	no	no	yes
1842 - Q850 Network Congestion	The network is congested. This code applies to ISDN or modem calls that came in over ISDN.	no	no	no	no	no	no	no	yes
1843 - Q850 Access Info Discarded	This code indicates that the network could not deliver access information to the remote user as requested. This code applies to ISDN or modem calls that came in over ISDN.	no	no	no	no	no	no	no	yes
1844 - Q850 Requested Channel Not Available	This code is returned when the circuit or channel that is indicated by the requesting entity cannot be provided by the other side of the interface. This code applies to ISDN or modem calls that came in over ISDN.	no	no	no	no	no	no	no	yes
1845 - Q850 Call Pre-empted	The call was preempted. This code applies to ISDN or modem calls that came in over ISDN.	no	no	no	no	no	no	no	yes
1847 - Q850 Resource Unavailable	This code is used to report a resource-unavailable event only when no other code in the resource-unavailable class applies. This code applies to ISDN or modem calls that came in over ISDN.	no	no	no	no	no	no	no	yes
1850 - Q850 Facility Not Subscribed	Not a subscribed facility. This code applies to ISDN or modem calls that came in over ISDN.	no	no	no	no	no	no	no	yes
1852 - Q850 Outgoing Call Barred	Although the calling party is a member of the closed user group for the outgoing closed user group call, outgoing calls are not allowed for this member. This code applies to ISDN or modem calls that came in over ISDN.	no	no	no	no	no	no	no	yes

Cause Codes	Description	11.0	11.1	11.2	11.3	12.0	12.1	12.2	12.3
Q850 Incoming Call Barred (1854)	Although the called party is a member of the closed user group for the incoming closed user group call, incoming calls are not allowed to this member. This code applies to ISDN or modem calls that have come in over ISDN.	no	no	no	no	no	no	no	yes
1858 - Q850 Bearer Capability Not Available	The user has requested a bearer capability that is implemented by the equipment that generated this code but that is not available at this time. This code applies to ISDN or modem calls that have come in over ISDN.	no	no	no	no	no	no	no	yes
1863 - Q850 Service Not Available	The code is used to report a service- or option-not-available event only when no other code in the service- or option-not-available class applies. This code applies to ISDN or modem calls that have come in over ISDN.	no	no	no	no	no	no	no	yes
1865 - Q850 Bearer Capability Not Implemented	The equipment that is sending this code does not support the bearer capability that was requested. This code applies to ISDN or modem calls that have come in over ISDN.	no	no	no	no	no	no	no	yes
1866 - Q850 Channel Not Implemented	The equipment that is sending this code does not support the channel type that was requested. This code applies to ISDN or modem calls that have come in over ISDN.	no	no	no	no	no	no	no	yes
1869 - Q850 Facility Not Implemented	The supplementary service requested by the user cannot be provided by the network. This code applies to ISDN or modem calls that have come in over ISDN.	no	no	no	no	no	no	no	yes
1881 - Q850 Invalid Call Reference	The equipment that is sending this code has received a message having a call reference that is not currently in use on the user-network interface. This code applies to ISDN or modem calls that have come in over ISDN.	no	no	no	no	no	no	no	yes
1882 - Q850 Channel Does Not Exist	The channel most recently identified is not acceptable to the sending entity for use in this call. This code applies to ISDN or modem calls that have come in over ISDN. This code applies to ISDN or modem calls that have come in over ISDN.	no	no	no	no	no	no	no	yes
1888 - Q850 Incompatible Destination	The equipment that is sending this code has received a request to establish a call that has low-layer compatibility or other compatibility attributes that cannot be accommodated. This code applies to ISDN or modem calls that have come in over ISDN.	no	no	no	no	no	no	no	yes
1896 - Q850 Mandatory Info Element Is Missing	The equipment that is sending this code has received a message that is missing an information element that must be present in the message before that message can be processed. This code applies to ISDN or modem calls that have come in over ISDN.	no	no	no	no	no	no	no	yes

Cause Codes	Description	11.0	11.1	11.2	11.3	12.0	12.1	12.2	12.3
1897 - Q850 Non Existent Message Type	The equipment that is sending this code has received a message with a message type that it does not recognize either because this is a message that is not defined or that is defined but not implemented by the equipment that is sending this code. This code applies to ISDN or modem calls that have come in over ISDN.	no	no	no	no	no	no	no	yes
1898 - Q850 Invalid Message	This code is used to report an invalid message when no other code in the invalid message class applies. This code applies to ISDN or modem calls that have come in over ISDN.	no	no	no	no	no	no	no	yes
1899 - Q850 Bad Info Element	The information element not recognized. This code applies to ISDN or modem calls that have come in over ISDN.	no	no	no	no	no	no	no	yes
1900 - Q850 Invalid Element Contents	The equipment that is sending this code has received an information element that it has implemented; however, one or more fields in the information element are coded in such a way that has not been implemented by the equipment that is sending this code. This code applies to ISDN or modem calls that have come in over ISDN.	no	no	no	no	no	no	no	yes
1901 - Q850 Wrong Message for State	The message that was received is incompatible with the call state. This code applies to ISDN or modem calls that have come in over ISDN.	no	no	no	no	no	no	no	yes
1902 - Q850 Recovery on Timer Expiration	A procedure has been initiated by the expiration of a timer in association with error-handling procedures. This code applies to ISDN or modem calls that have come in over ISDN.	no	no	no	no	no	no	no	yes
1903 - Q850 Info Element Error	The equipment that is sending this code has received a message that includes information elements or parameters that are not recognized because the information element identifiers or parameter names are not defined or are defined but not implemented by the equipment that is sending this code. This code applies to ISDN or modem calls that have come in over ISDN.	no	no	no	no	no	no	no	yes
1911 - Q850 Protocol Error	This code is used to report a protocol error event only when no other code in the protocol error class applies. This code applies to ISDN or modem calls that have come in over ISDN.	no	no	no	no	no	no	no	yes
1927 - Q850 Unspecified Internetworking Event	There has been an error when interworking with a network that does not provide codes for actions that it takes. This code applies to ISDN or modem calls that have come in over ISDN.	no	no	no	no	no	no	no	yes

Configuring AAA Server Group Selection Based on DNIS

Cisco software allows you to authenticate users to a particular AAA server group based on the Dialed Number Identification Service (DNIS) number of the session. Any phone line (a regular home phone or a commercial

T1/PRI line) can be associated with several phone numbers. The DNIS number identifies the number that was called to reach you.

For example, suppose you want to share the same phone number with several customers, but you want to know which customer is calling before you pick up the phone. You can customize how you answer the phone because DNIS allows you to know which customer is calling when you answer.

Cisco devices with either ISDN or internal modems can receive the DNIS number. This functionality allows users to assign different TACACS+ server groups for different customers (that is, different TACACS+ servers for different DNIS numbers). Additionally, using server groups you can specify the same server group for AAA services or a separate server group for each AAA service.

Cisco IOS software provides the flexibility to implement authentication and accounting services in several ways:

- Globally--AAA services are defined using global configuration access list commands and applied in general to all interfaces on a specific network access server.
- Per interface--AAA services are defined using interface configuration commands and applied specifically to the interface being configured on a specific network access server.
- DNIS mapping--You can use DNIS to specify an AAA server to supply AAA services.

Because AAA configuration methods can be configured simultaneously, Cisco has established an order of precedence to determine which server or groups of servers provide AAA services. The order of precedence is as follows:

- Per DNIS--If you configure the network access server to use DNIS to identify which server group provides AAA services, then this method takes precedence over any additional AAA selection method.
- Per interface--If you configure the network access server per interface to use access lists to determine how a server provides AAA services, this method takes precedence over any global configuration AAA access lists.
- Globally--If you configure the network access server by using global AAA access lists to determine how the security server provides AAA services, this method has the lowest precedence.



Note Prior to configuring AAA Server Group Selection Based on DNIS, you must configure the remote security servers associated with each AAA server group. See *Identifying the TACACS Server Host* and *Configuring AAA Server Groups* for more information.

To configure the device to select a particular AAA server group based on the DNIS of the server group, configure DNIS mapping. To map a server group with a group name with DNIS number, use the following commands in global configuration mode:

SUMMARY STEPS

1. Device>**enable**
2. Device#**configure terminal**
3. Device (config)#**aaa dnis map enable**
4. Router(config)# **aaa dnis map** *dnis-number* **authentication ppp group** *server-group-name*
5. Router(config)# **aaa dnis map** *dnis-number* **accounting network** [**none** | **start-stop** | **stop-only**] **group** *server-group-name*

DETAILED STEPS

	Command or Action	Purpose
Step 1	Device>enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	Device#configure terminal	Enters global configuration mode.
Step 3	Device (config)#aaa dnis map enable	Enables DNIS mapping.
Step 4	Router(config)# aaa dnis map <i>dnis-number</i> authentication ppp group <i>server-group-name</i>	Maps a DNIS number to a defined AAA server group; the servers in this server group are being used for authentication.
Step 5	Router(config)# aaa dnis map <i>dnis-number</i> accounting network [none start-stop stop-only] group <i>server-group-name</i>	Maps a DNIS number to a defined AAA server group; the servers in this server group are being used for accounting.

TACACS+ Configuration Options

You can configure the switch to use a single server or AAA server groups to group existing server hosts for authentication. You can group servers to select a subset of the configured server hosts and use them for a particular service. The server group is used with a global server-host list and contains the list of IP addresses of the selected server hosts.

TACACS+ Login Authentication

A method list describes the sequence and authentication methods to be queried to authenticate a user. You can designate one or more security protocols to be used for authentication, thus ensuring a backup system for authentication in case the initial method fails. The software uses the first method listed to authenticate users; if that method fails to respond, the software selects the next authentication method in the method list. This process continues until there is successful communication with a listed authentication method or until all defined methods are exhausted. If authentication fails at any point in this cycle—meaning that the security server or local username database responds by denying the user access—the authentication process stops, and no other authentication methods are attempted.

TACACS+ Authorization for Privileged EXEC Access and Network Services

AAA authorization limits the services available to a user. When AAA authorization is enabled, the switch uses information retrieved from the user's profile, which is located either in the local user database or on the security server, to configure the user's session. The user is granted access to a requested service only if the information in the user profile allows it.

TACACS+ Authentication

After you have identified the TACACS+ daemon and defined an associated TACACS+ encryption key, you must define method lists for TACACS+ authentication. Because TACACS+ authentication is operated via AAA, you need to issue the **aaa authentication** command, specifying TACACS+ as the authentication method.

TACACS+ Authorization

AAA authorization enables you to set parameters that restrict a user's access to the network. Authorization via TACACS+ may be applied to commands, network connections, and EXEC sessions. Because TACACS+ authorization is facilitated through AAA, you must issue the **aaa authorization** command, specifying TACACS+ as the authorization method.

TACACS+ Accounting

The AAA accounting feature tracks the services that users are accessing and the amount of network resources that they are consuming. When AAA accounting is enabled, the switch reports user activity to the TACACS+ security server in the form of accounting records. Each accounting record contains accounting attribute-value (AV) pairs and is stored on the security server. This data can then be analyzed for network management, client billing, or auditing.

Default TACACS+ Configuration

TACACS+ and AAA are disabled by default.

To prevent a lapse in security, you cannot configure TACACS+ through a network management application. When enabled, TACACS+ can authenticate users accessing the switch through the CLI.



Note Although TACACS+ configuration is performed through the CLI, the TACACS+ server authenticates HTTP connections that have been configured with a privilege level of 15.

Per VRF for TACACS Servers

The Per VRF for TACACS+ Servers feature allows per virtual routing and forwarding (VRF) AAA to be configured on TACACS+ servers. TACACS+ server access is required to configure this feature.

How to Configure TACACS+

Identifying the TACACS+ Server Host and Setting the Authentication Key

Follow these steps to identify the TACACS+ server host and set the authentication key:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **tacacs-server host *hostname***
4. **aaa new-model**
5. **aaa group server tacacs+ *group-name***
6. **server *ip-address***

7. `end`
8. `show running-config`
9. `copy running-config startup-config`

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> <code>enable</code>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode.
Step 3	tacacs-server host <i>hostname</i> Example: Device(config)# <code>tacacs-server host yourserver</code>	Identifies the IP host or hosts maintaining a TACACS+ server. Enter this command multiple times to create a list of preferred hosts. The software searches for hosts in the order in which you specify them. For <i>hostname</i> , specify the name or IP address of the host.
Step 4	aaa new-model Example: Device(config)# <code>aaa new-model</code>	Enables AAA.
Step 5	aaa group server tacacs+ <i>group-name</i> Example: Device(config)# <code>aaa group server tacacs+ your_server_group</code>	(Optional) Defines the AAA server-group with a group name. This command puts the Device in a server group subconfiguration mode.
Step 6	server <i>ip-address</i> Example: Device(config)# <code>server 10.1.2.3</code>	(Optional) Associates a particular TACACS+ server with the defined server group. Repeat this step for each TACACS+ server in the AAA server group. Each server in the group must be previously defined in Step 3.
Step 7	end Example: Device(config)# <code>end</code>	Returns to privileged EXEC mode.

	Command or Action	Purpose
Step 8	show running-config Example: Device# <code>show running-config</code>	Verifies your entries.
Step 9	copy running-config startup-config Example: Device# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Configuring TACACS+ Login Authentication

Follow these steps to configure TACACS+ login authentication:

Before you begin

To configure AAA authentication, you define a named list of authentication methods and then apply that list to various ports.



Note To secure the for HTTP access by using AAA methods, you must configure the with the **ip http authentication aaa** global configuration command. Configuring AAA authentication does not secure the for HTTP access by using AAA methods.

For more information about the **ip http authentication** command, see the *Cisco IOS Security Command Reference, Release 12.4*.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **aaa new-model**
4. **aaa authentication login {default | list-name} method1 [method2...]**
5. **line [console | tty | vty] line-number [ending-line-number]**
6. **login authentication {default | list-name}**
7. **end**
8. **show running-config**
9. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable	Enables privileged EXEC mode.

	Command or Action	Purpose
	<p>Example:</p> <pre>Device> enable</pre>	<ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	<p>configure terminal</p> <p>Example:</p> <pre>Device# configure terminal</pre>	Enters global configuration mode.
Step 3	<p>aaa new-model</p> <p>Example:</p> <pre>Device(config)# aaa new-model</pre>	Enables AAA.
Step 4	<p>aaa authentication login {default list-name} method1 [method2...]</p> <p>Example:</p> <pre>Device(config)# aaa authentication login default tacacs+ local</pre>	<p>Creates a login authentication method list.</p> <ul style="list-style-type: none"> • To create a default list that is used when a named list is <i>not</i> specified in the login authentication command, use the default keyword followed by the methods that are to be used in default situations. The default method list is automatically applied to all ports. • For <i>list-name</i>, specify a character string to name the list you are creating. • For <i>method1...</i>, specify the actual method the authentication algorithm tries. The additional methods of authentication are used only if the previous method returns an error, not if it fails. <p>Select one of these methods:</p> <ul style="list-style-type: none"> • <i>enable</i>—Use the enable password for authentication. Before you can use this authentication method, you must define an enable password by using the enable password global configuration command. • <i>group tacacs+</i>—Uses TACACS+ authentication. Before you can use this authentication method, you must configure the TACACS+ server. • <i>line</i>—Use the line password for authentication. Before you can use this authentication method, you must define a line password. Use the password password line configuration command. • <i>local</i>—Use the local username database for authentication. You must enter username information in the database. Use the username password global configuration command.

	Command or Action	Purpose
		<ul style="list-style-type: none"> • <i>local-case</i>—Use a case-sensitive local username database for authentication. You must enter username information in the database by using the username name password global configuration command. • <i>none</i>—Do not use any authentication for login.
Step 5	line [console tty vty] <i>line-number</i> [<i>ending-line-number</i>] Example: <pre>Device(config)# line 2 4</pre>	Enters line configuration mode, and configures the lines to which you want to apply the authentication list.
Step 6	login authentication {default <i>list-name</i> } Example: <pre>Device(config-line)# login authentication default</pre>	Applies the authentication list to a line or set of lines. <ul style="list-style-type: none"> • If you specify default, use the default list created with the aaa authentication login command. • For <i>list-name</i>, specify the list created with the aaa authentication login command.
Step 7	end Example: <pre>Device(config-line)# end</pre>	Returns to privileged EXEC mode.
Step 8	show running-config Example: <pre>Device# show running-config</pre>	Verifies your entries.
Step 9	copy running-config startup-config Example: <pre>Device# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Configuring TACACS+ Authorization for Privileged EXEC Access and Network Services

You can use the **aaa authorization** global configuration command with the **tacacs+** keyword to set parameters that restrict a user's network access to privileged EXEC mode.



Note Authorization is bypassed for authenticated users who log in through the CLI even if authorization has been configured.

Follow these steps to specify TACACS+ authorization for privileged EXEC access and network services:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **aaa authorization network tacacs+**
4. **aaa authorization exec tacacs+**
5. **end**
6. **show running-config**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	aaa authorization network tacacs+ Example: Device(config)# aaa authorization network tacacs+	Configures the switch for user TACACS+ authorization for all network-related service requests.
Step 4	aaa authorization exec tacacs+ Example: Device(config)# aaa authorization exec tacacs+	Configures the switch for user TACACS+ authorization if the user has privileged EXEC access. The exec keyword might return user profile information (such as autocommand information).
Step 5	end Example: Device(config)# end	Returns to privileged EXEC mode.

	Command or Action	Purpose
Step 6	show running-config Example: <pre>Device# show running-config</pre>	Verifies your entries.
Step 7	copy running-config startup-config Example: <pre>Device# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Starting TACACS+ Accounting

Follow these steps to start TACACS+ Accounting:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **aaa accounting network start-stop tacacs+**
4. **aaa accounting exec start-stop tacacs+**
5. **end**
6. **show running-config**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Device> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: <pre>Device# configure terminal</pre>	Enters global configuration mode.
Step 3	aaa accounting network start-stop tacacs+ Example: <pre>Device(config)# aaa accounting network start-stop tacacs+</pre>	Enables TACACS+ accounting for all network-related service requests.

	Command or Action	Purpose
Step 4	aaa accounting exec start-stop tacacs+ Example: Device(config)# aaa accounting exec start-stop tacacs+	Enables TACACS+ accounting to send a start-record accounting notice at the beginning of a privileged EXEC process and a stop-record at the end.
Step 5	end Example: Device(config)# end	Returns to privileged EXEC mode.
Step 6	show running-config Example: Device# show running-config	Verifies your entries.
Step 7	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

What to do next

To establish a session with a router if the AAA server is unreachable, use the **aaa accounting system guarantee-first** command. It guarantees system accounting as the first record, which is the default condition. In some situations, users might be prevented from starting a session on the console or terminal connection until after the system reloads, which can take more than 3 minutes.

To establish a console or Telnet session with the router if the AAA server is unreachable when the router reloads, use the **no aaa accounting system guarantee-first** command.

Establishing a Session with a Router if the AAA Server is Unreachable

To establishing a session with a router if the AAA server is unreachable, use the **aaa accounting system guarantee-first** command. It guarantees system accounting as the first record, which is the default condition. In some situations, users might be prevented from starting a session on the console or terminal connection until after the system reloads, which can take more than 3 minutes.

To establish a console or Telnet session with the router if the AAA server is unreachable when the router reloads, use the **no aaa accounting system guarantee-first** command.

Establishing a Session with a Router if the AAA Server is Unreachable

The **aaa accounting system guarantee-first** command guarantees system accounting as the first record, which is the default condition. In some situations, users might be prevented from starting a session on the console or terminal connection until after the system reloads, which can take more than 3 minutes.

To establish a console or Telnet session with the router if the AAA server is unreachable when the router reloads, use the **no aaa accounting system guarantee-first** command.

Configuring Per VRF on a TACACS Server

The initial steps in this procedure are used to configure AAA and a server group, create a VRF routing table, and configure an interface. Steps 10 through 13 are used to configure the per VRF on a TACACS+ server feature:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip vrf** *vrf-name*
4. **rd** *route-distinguisher*
5. **exit**
6. **interface** *interface-name*
7. **ip vrf forwarding** *vrf-name*
8. **ip address** *ip-address mask* [**secondary**]
9. **exit**
10. **aaa group server tacacs+** *group-name*
11. **server-private** {*ip-address* | *name*} [**nat**] [**single-connection**] [**port** *port-number*] [**timeout** *seconds*] [**key** [**0** | **7**] *string*]
12. **ip vrf forwarding** *vrf-name*
13. **ip tacacs source-interface** *subinterface-name*
14. **exit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	ip vrf <i>vrf-name</i> Example:	Configures a VRF table and enters VRF configuration mode.

	Command or Action	Purpose
	Device(config)# ip vrf cisco	
Step 4	rd <i>route-distinguisher</i> Example: Device(config-vrf)# rd 100:1	Creates routing and forwarding tables for a VRF instance.
Step 5	exit Example: Device(config-vrf)# exit	Exits VRF configuration mode.
Step 6	interface <i>interface-name</i> Example: Device(config)# interface Loopback0	Configures an interface and enters interface configuration mode.
Step 7	ip vrf forwarding <i>vrf-name</i> Example: Device(config-if)# ip vrf forwarding cisco	Configures a VRF for the interface.
Step 8	ip address <i>ip-address mask [secondary]</i> Example: Device(config-if)# ip address 10.0.0.2 255.0.0.0	Sets a primary or secondary IP address for an interface.
Step 9	exit Example: Device(config-if)# exit	Exits interface configuration mode.
Step 10	aaa group server tacacs+ <i>group-name</i> Example: Device(config)# aaa group server tacacs+ tacacs1	Groups different TACACS+ server hosts into distinct lists and distinct methods and enters server-group configuration mode.
Step 11	server-private { <i>ip-address name</i> } [nat] [single-connection] [port <i>port-number</i>] [timeout <i>seconds</i>] [key [0 7] <i>string</i>] Example: Device(config-sg-tacacs+)# server-private 10.1.1.1 port 19 key cisco	Configures the IP address of the private TACACS+ server for the group server.
Step 12	ip vrf forwarding <i>vrf-name</i> Example:	Configures the VRF reference of a AAA TACACS+ server group.

	Command or Action	Purpose
	Device(config-sg-tacacs)# ip vrf forwarding cisco	
Step 13	ip tacacs source-interface <i>subinterface-name</i> Example: Device(config-sg-tacacs)# ip tacacs source-interface Loopback0	Uses the IP address of a specified interface for all outgoing TACACS+ packets.
Step 14	exit Example: Device(config-sg-tacacs)# exit	Exits server-group configuration mode.

Verifying Per VRF for TACACS Servers

To verify the per VRF TACACS+ configuration, perform the following steps:



Note The **debug** commands may be used in any order.

SUMMARY STEPS

1. enable
2. debug tacacs authentication
3. debug tacacs authorization
4. debug tacacs accounting
5. debug tacacs packets

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	debug tacacs authentication Example: Device# debug tacacs authentication	Displays information about AAA/TACACS+ authentication.
Step 3	debug tacacs authorization Example:	Displays information about AAA/TACACS+ authorization.

	Command or Action	Purpose
	Device# debug tacacs authorization	
Step 4	debug tacacs accounting Example: Device# debug tacacs accounting	Displays information about accountable events as they occur.
Step 5	debug tacacs packets Example: Device# debug tacacs packets	Displays information about TACACS+ packets.

Monitoring TACACS+

Table 109: Commands for Displaying TACACS+ Information

Command	Purpose
show tacacs	Displays TACACS+ server statistics.

Configuration Examples for TACACS+

Example: TACACS Authorization

The following example shows how to configure TACACS+ as the security protocol for PPP authentication using the default method list; it also shows how to configure network authorization via TACACS+:

```
aaa new-model
aaa authentication ppp default if-needed group tacacs+ local
aaa authorization network default group tacacs+
tacacs-server host 10.1.2.3
tacacs-server key goaway
interface serial 0
  ppp authentication chap default
```

The lines in the preceding sample configuration are defined as follows:

- The **aaa new-model** command enables the AAA security services.
- The **aaa authentication** command defines a method list, “default,” to be used on serial interfaces running PPP. The keyword **default** means that PPP authentication is applied by default to all interfaces. The **if-needed** keyword means that if the user has already authenticated by going through the ASCII login procedure, then PPP authentication is not necessary and can be skipped. If authentication is needed, the keyword **group tacacs+** means that authentication will be done through TACACS+. If TACACS+ returns an ERROR of some sort during authentication, the keyword **local** indicates that authentication will be attempted using the local database on the network access server.

- The **aaa authorization** command configures network authorization via TACACS+. Unlike authentication lists, this authorization list always applies to all incoming network connections made to the network access server.
- The **tacacs-server host** command identifies the TACACS+ daemon as having an IP address of 10.1.2.3. The **tacacs-server key** command defines the shared encryption key to be “goaway.”
- The **interface** command selects the line, and the **ppp authentication** command applies the default method list to this line.

Example: TACACS Accounting

The following example shows how to configure TACACS+ as the security protocol for PPP authentication using the default method list; it also shows how to configure accounting via TACACS+:

```
aaa new-model
aaa authentication ppp default if-needed group tacacs+ local
aaa accounting network default stop-only group tacacs+
tacacs-server host 10.1.2.3
tacacs-server key goaway
interface serial 0
  ppp authentication chap default
```

The lines in the preceding sample configuration are defined as follows:

- The **aaa new-model** command enables the AAA security services.
- The **aaa authentication** command defines a method list, “default,” to be used on serial interfaces running PPP. The keyword **default** means that PPP authentication is applied by default to all interfaces. The **if-needed** keyword means that if the user has already authenticated by going through the ASCII login procedure, then PPP authentication is not necessary and can be skipped. If authentication is needed, the keyword **group tacacs+** means that authentication will be done through TACACS+. If TACACS+ returns an ERROR of some sort during authentication, the keyword **local** indicates that authentication will be attempted using the local database on the network access server.
- The **aaa accounting** command configures network accounting via TACACS+. In this example, accounting records describing the session that just terminated will be sent to the TACACS+ daemon whenever a network connection terminates.
- The **tacacs-server host** command identifies the TACACS+ daemon as having an IP address of 10.1.2.3. The **tacacs-server key** command defines the shared encryption key to be “goaway.”
- The **interface** command selects the line, and the **ppp authentication** command applies the default method list to this line.

Example: TACACS Authentication

The following example shows how to configure TACACS+ as the security protocol for PPP authentication:

```
aaa new-model
aaa authentication ppp test group tacacs+ local
tacacs-server host 10.1.2.3
tacacs-server key goaway
```

```
interface serial 0
  ppp authentication chap pap test
```

The lines in the preceding sample configuration are defined as follows:

- The **aaa new-model** command enables the AAA security services.
- The **aaa authentication** command defines a method list, “test,” to be used on serial interfaces running PPP. The keyword **group tacacs+** means that authentication will be done through TACACS+. If TACACS+ returns an ERROR of some sort during authentication, the keyword **local** indicates that authentication will be attempted using the local database on the network access server.
- The **tacacs-server host** command identifies the TACACS+ daemon as having an IP address of 10.1.2.3. The **tacacs-server key** command defines the shared encryption key to be “goaway.”
- The **interface** command selects the line, and the **ppp authentication** command applies the test method list to this line.

The following example shows how to configure TACACS+ as the security protocol for PPP authentication, but instead of the “test” method list, the “default” method list is used.

```
aaa new-model
aaa authentication ppp default if-needed group tacacs+ local
tacacs-server host 10.1.2.3
tacacs-server key goaway
interface serial 0
  ppp authentication chap default
```

The lines in the preceding sample configuration are defined as follows:

- The **aaa new-model** command enables the AAA security services.
- The **aaa authentication** command defines a method list, “default,” to be used on serial interfaces running PPP. The keyword **default** means that PPP authentication is applied by default to all interfaces. The **if-needed** keyword means that if the user has already authenticated by going through the ASCII login procedure, then PPP authentication is not necessary and can be skipped. If authentication is needed, the keyword **group tacacs+** means that authentication will be done through TACACS+. If TACACS+ returns an ERROR of some sort during authentication, the keyword **local** indicates that authentication will be attempted using the local database on the network access server.
- The **tacacs-server host** command identifies the TACACS+ daemon as having an IP address of 10.1.2.3. The **tacacs-server key** command defines the shared encryption key to be “goaway.”
- The **interface** command selects the line, and the **ppp authentication** command applies the default method list to this line.

The following example shows how to create the same authentication algorithm for PAP, but it calls the method list “MIS-access” instead of “default”:

```
aaa new-model
aaa authentication pap MIS-access if-needed group tacacs+ local
tacacs-server host 10.1.2.3
tacacs-server key goaway
interface serial 0
  ppp authentication pap MIS-access
```


The lines in the preceding sample configuration are defined as follows:

- The **aaa new-model** command enables the AAA security services.
- The **aaa authentication** command defines a method list, “MIS-access,” to be used on serial interfaces running PPP. The method list, “MIS-access,” means that PPP authentication is applied to all interfaces. The **if-needed** keyword means that if the user has already authenticated by going through the ASCII login procedure, then PPP authentication is not necessary and can be skipped. If authentication is needed, the keyword **group tacacs+** means that authentication will be done through TACACS+. If TACACS+ returns an ERROR of some sort during authentication, the keyword **local** indicates that authentication will be attempted using the local database on the network access server.
- The **tacacs-server host** command identifies the TACACS+ daemon as having an IP address of 10.1.2.3. The **tacacs-server key** command defines the shared encryption key to be “goaway.”
- The **interface** command selects the line, and the **ppp authentication** command applies the default method list to this line.

The following example shows the configuration for a TACACS+ daemon with an IP address of 10.2.3.4 and an encryption key of “apple”:

```
aaa new-model
aaa authentication login default group tacacs+ local
tacacs-server host 10.2.3.4
tacacs-server key apple
```

The lines in the preceding sample configuration are defined as follows:

- The **aaa new-model** command enables the AAA security services.
- The **aaa authentication** command defines the default method list. Incoming ASCII logins on all interfaces (by default) will use TACACS+ for authentication. If no TACACS+ server responds, then the network access server will use the information contained in the local username database for authentication.
- The **tacacs-server host** command identifies the TACACS+ daemon as having an IP address of 10.2.3.4. The **tacacs-server key** command defines the shared encryption key to be “apple.”

Example: Configuring Per VRF for TACACS Servers

The following output example shows that the group server **tacacs1** is configured for per VRF AAA services:

```
aaa group server tacacs+ tacacs1
  server-private 10.1.1.1 port 19 key cisco
  ip vrf forwarding cisco
  ip tacacs source-interface Loopback0
ip vrf cisco
  rd 100:1
interface Loopback0
  ip address 10.0.0.2 255.0.0.0
  ip vrf forwarding cisco
```

Additional References for TACACS+

Related Documents

Related Topic	Document Title
Cisco security commands	<ul style="list-style-type: none"> • Cisco IOS Security Command Reference: Commands A to C • Cisco IOS Security Command Reference: Commands D to L • Cisco IOS Security Command Reference: Commands M to R • Cisco IOS Security Command Reference: Commands S to Z
IPv6 commands	Cisco IOS IPv6 Command Reference

MIBs

MB	MIBs Link
	<p>To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL:</p> <p>http://www.cisco.com/go/mibs</p>

Technical Assistance

Description	Link
<p>The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies.</p> <p>To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds.</p> <p>Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.</p>	http://www.cisco.com/support

Feature Information for TACACS+

Release	Feature Information
Cisco IOS Release 15.0(2)EX1	This feature was introduced.

Release	Feature Information
Cisco IOS 12.2(54)SG Cisco IOS 15.2(1)E	<p>The Per VRF for TACACS+ Servers feature allows per virtual route forwarding (per VRF) to be configured for authentication, authorization, and accounting (AAA) on TACACS+ servers.</p> <p>The following commands were introduced or modified: ip tacacs source-interface, ip vrf forwarding (server-group), server-private (TACACS+).</p>



CHAPTER 51

Configuring RADIUS

The RADIUS security system is a distributed client/server system that secures networks against unauthorized access. In the Cisco implementation, RADIUS clients run on Cisco devices and send authentication requests to a central RADIUS server that contains all user authentication and network service access information.

- [Prerequisites for Configuring RADIUS, on page 1059](#)
- [Restrictions for Configuring RADIUS, on page 1060](#)
- [Information about RADIUS, on page 1060](#)
- [How to Configure RADIUS, on page 1080](#)
- [Configuration Examples for RADIUS, on page 1095](#)
- [Additional References for RADIUS, on page 1098](#)
- [Feature Information for RADIUS, on page 1099](#)

Prerequisites for Configuring RADIUS

This section lists the prerequisites for controlling Device access with RADIUS.

General:

- RADIUS and Authentication, Authorization, and Accounting (AAA) must be enabled to use any of the configuration commands in this chapter.
- RADIUS is facilitated through AAA and can be enabled only through AAA commands.
- Use the **aaa new-model** global configuration command to enable AAA.
- Use the **aaa authentication** global configuration command to define method lists for RADIUS authentication.
- Use **line** and **interface** commands to enable the defined method lists to be used.
- At a minimum, you must identify the host or hosts that run the RADIUS server software and define the method lists for RADIUS authentication. You can optionally define method lists for RADIUS authorization and accounting.
- You should have access to and should configure a RADIUS server before configuring RADIUS features on your Device.
- The RADIUS host is normally a multiuser system running RADIUS server software from Cisco (Cisco Secure Access Control Server Version 3.0), Livingston, Merit, Microsoft, or another software provider. For more information, see the RADIUS server documentation.

- To use the Change-of-Authorization (CoA) interface, a session must already exist on the switch. CoA can be used to identify a session and enforce a disconnect request. The update affects only the specified session.

For RADIUS operation:

- Users must first successfully complete RADIUS authentication before proceeding to RADIUS authorization, if it is enabled.

Restrictions for Configuring RADIUS

This topic covers restrictions for controlling Device access with RADIUS.

General:

- To prevent a lapse in security, you cannot configure RADIUS through a network management application.

RADIUS is not suitable in the following network security situations:

- Multiprotocol access environments. RADIUS does not support AppleTalk Remote Access (ARA), NetBIOS Frame Control Protocol (NBFCP), NetWare Asynchronous Services Interface (NASI), or X.25 PAD connections.
- Switch-to-switch or router-to-router situations. RADIUS does not provide two-way authentication. RADIUS can be used to authenticate from one device to a non-Cisco device if the non-Cisco device requires authentication.
- Networks using a variety of services. RADIUS generally binds a user to one service model.

Information about RADIUS

RADIUS and Switch Access

This section describes how to enable and configure RADIUS. RADIUS provides detailed accounting information and flexible administrative control over the authentication and authorization processes.

RADIUS Overview

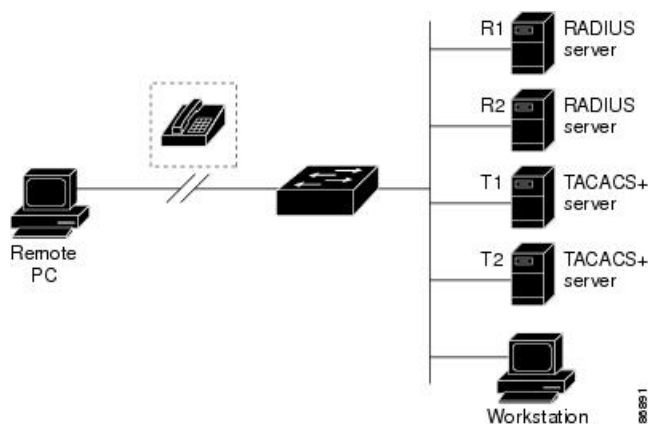
RADIUS is a distributed client/server system that secures networks against unauthorized access. RADIUS clients run on supported Cisco routers and switches. Clients send authentication requests to a central RADIUS server, which contains all user authentication and network service access information.

Use RADIUS in these network environments that require access security:

- Networks with multiple-vendor access servers, each supporting RADIUS. For example, access servers from several vendors use a single RADIUS server-based security database. In an IP-based network with multiple vendors' access servers, dial-in users are authenticated through a RADIUS server that has been customized to work with the Kerberos security system.

- Turnkey network security environments in which applications support the RADIUS protocol, such as in an access environment that uses a *smart card* access control system. In one case, RADIUS has been used with Enigma's security cards to validate users and to grant access to network resources.
- Networks already using RADIUS. You can add a Cisco Device containing a RADIUS client to the network. This might be the first step when you make a transition to a TACACS+ server. See Figure 2: Transitioning from RADIUS to TACACS+ Services below.
- Network in which the user must only access a single service. Using RADIUS, you can control user access to a single host, to a single utility such as Telnet, or to the network through a protocol such as IEEE 802.1x. For more information about this protocol, see Chapter 11, "Configuring IEEE 802.1x Port-Based Authentication."
- Networks that require resource accounting. You can use RADIUS accounting independently of RADIUS authentication or authorization. The RADIUS accounting functions allow data to be sent at the start and end of services, showing the amount of resources (such as time, packets, bytes, and so forth) used during the session. An Internet service provider might use a freeware-based version of RADIUS access control and accounting software to meet special security and billing needs.

Figure 86: Transitioning from RADIUS to TACACS+ Services



RADIUS Operation

When a user attempts to log in and authenticate to a Device that is access controlled by a RADIUS server, these events occur:

1. The user is prompted to enter a username and password.
2. The username and encrypted password are sent over the network to the RADIUS server.
3. The user receives one of the following responses from the RADIUS server:
 - ACCEPT—The user is authenticated.
 - REJECT—The user is either not authenticated and is prompted to re-enter the username and password, or access is denied.
 - CHALLENGE—A challenge requires additional data from the user.
 - CHALLENGE PASSWORD—A response requests the user to select a new password.

The ACCEPT or REJECT response is bundled with additional data that is used for privileged EXEC or network authorization. The additional data included with the ACCEPT or REJECT packets includes these items:

- Telnet, SSH, rlogin, or privileged EXEC services
- Connection parameters, including the host or client IP address, access list, and user timeouts

Default RADIUS Configuration

RADIUS and AAA are disabled by default.

To prevent a lapse in security, you cannot configure RADIUS through a network management application. When enabled, RADIUS can authenticate users accessing the switch through the CLI.

RADIUS Server Host

Switch-to-RADIUS-server communication involves several components:

- Hostname or IP address
- Authentication destination port
- Accounting destination port
- Key string
- Timeout period
- Retransmission value

You identify RADIUS security servers by their hostname or IP address, hostname and specific UDP port numbers, or their IP address and specific UDP port numbers. The combination of the IP address and the UDP port number creates a unique identifier, allowing different ports to be individually defined as RADIUS hosts providing a specific AAA service. This unique identifier enables RADIUS requests to be sent to multiple UDP ports on a server at the same IP address.

If two different host entries on the same RADIUS server are configured for the same service—for example, accounting—the second host entry configured acts as a fail-over backup to the first one. Using this example, if the first host entry fails to provide accounting services, the %RADIUS-4-RADIUS_DEAD message appears, and then the switch tries the second host entry configured on the same device for accounting services. (The RADIUS host entries are tried in the order that they are configured.)

A RADIUS server and the switch use a shared secret text string to encrypt passwords and exchange responses. To configure RADIUS to use the AAA security commands, you must specify the host running the RADIUS server daemon and a secret text (key) string that it shares with the switch.

The timeout, retransmission, and encryption key values can be configured globally for all RADIUS servers, on a per-server basis, or in some combination of global and per-server settings.

RADIUS Login Authentication

To configure AAA authentication, you define a named list of authentication methods and then apply that list to various ports. The method list defines the types of authentication to be performed and the sequence in which

they are performed; it must be applied to a specific port before any of the defined authentication methods are performed. The only exception is the default method list. The default method list is automatically applied to all ports except those that have a named method list explicitly defined.

A method list describes the sequence and authentication methods to be queried to authenticate a user. You can designate one or more security protocols to be used for authentication, thus ensuring a backup system for authentication in case the initial method fails. The software uses the first method listed to authenticate users; if that method fails to respond, the software selects the next authentication method in the method list. This process continues until there is successful communication with a listed authentication method or until all defined methods are exhausted. If authentication fails at any point in this cycle—meaning that the security server or local username database responds by denying the user access—the authentication process stops, and no other authentication methods are attempted.

AAA Server Groups

You can configure the switch to use AAA server groups to group existing server hosts for authentication. You select a subset of the configured server hosts and use them for a particular service. The server group is used with a global server-host list, which lists the IP addresses of the selected server hosts.

Server groups also can include multiple host entries for the same server if each entry has a unique identifier (the combination of the IP address and UDP port number), allowing different ports to be individually defined as RADIUS hosts providing a specific AAA service. This unique identifier enables RADIUS requests to be sent to different UDP ports on a server at the same IP address. If you configure two different host entries on the same RADIUS server for the same service, (for example, accounting), the second configured host entry acts as a fail-over backup to the first one. If the first host entry fails to provide accounting services, the network access server tries the second host entry configured on the same device for accounting services. (The RADIUS host entries are tried in the order in which they are configured.)

AAA Authorization

AAA authorization limits the services available to a user. When AAA authorization is enabled, the switch uses information retrieved from the user's profile, which is in the local user database or on the security server, to configure the user's session. The user is granted access to a requested service only if the information in the user profile allows it.

RADIUS Accounting

The AAA accounting feature tracks the services that users are using and the amount of network resources that they are consuming. When you enable AAA accounting, the switch reports user activity to the RADIUS security server in the form of accounting records. Each accounting record contains accounting attribute-value (AV) pairs and is stored on the security server. You can then analyze the data for network management, client billing, or auditing.

Vendor-Specific RADIUS Attributes

The Internet Engineering Task Force (IETF) draft standard specifies a method for communicating vendor-specific information between the switch and the RADIUS server by using the vendor-specific attribute (attribute 26). Vendor-specific attributes (VSAs) allow vendors to support their own extended attributes not suitable for general use. The Cisco RADIUS implementation supports one vendor-specific option by using

the format recommended in the specification. Cisco's vendor-ID is 9, and the supported option has vendor-type 1, which is named *cisco-avpair*. The value is a string with this format:

```
protocol : attribute sep value *
```

Protocol is a value of the Cisco protocol attribute for a particular type of authorization. *Attribute* and *value* are an appropriate attributevalue (AV) pair defined in the Cisco TACACS+ specification, and *sep* is = for mandatory attributes and is * for optional attributes. The full set of features available for TACACS+ authorization can then be used for RADIUS.

For example, the following AV pair causes Cisco's "multiple named IP address pools" feature to be activated during IP authorization (during PPP's Internet Protocol Control Protocol (IPCP) address assignment):

```
cisco-avpair= "ip:addr-pool=first"
```

If you insert an "*", the AV pair "ip:addr-pool=first" becomes optional. Note that any AV pair can be made optional:

```
cisco-avpair= "ip:addr-pool*first"
```

The following example shows how to cause a user logging in from a network access server to have immediate access to EXEC commands:

```
cisco-avpair= "shell:priv-lvl=15"
```

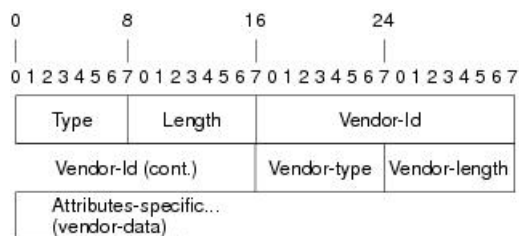
Other vendors have their own unique vendor-IDs, options, and associated VSAs. For more information about vendor-IDs and VSAs, see RFC 2138, "Remote Authentication Dial-In User Service (RADIUS)."

Attribute 26 contains the following three elements:

- Type
- Length
- String (also known as data)
 - Vendor-Id
 - Vendor-Type
 - Vendor-Length
 - Vendor-Data

The figure below shows the packet format for a VSA encapsulated "behind" attribute 26.

Figure 87: VSA Encapsulated Behind Attribute 26



51325



Note It is up to the vendor to specify the format of their VSA. The Attribute-Specific field (also known as Vendor-Data) is dependent on the vendor's definition of that attribute.

The table below describes significant fields listed in the Vendor-Specific RADIUS IETF Attributes table (second table below), which lists supported vendor-specific RADIUS attributes (IETF attribute 26).

Table 110: Vendor-Specific Attributes Table Field Descriptions

Field	Description
Number	All attributes listed in the following table are extensions of IETF attribute 26.
Vendor-Specific Command Codes	A defined code used to identify a particular vendor. Code 9 defines Cisco VSAs, 311 defines Microsoft VSAs, and 529 defines Ascend VSAs.
Sub-Type Number	The attribute ID number. This number is much like the ID numbers of IETF attributes, except it is a "second layer" ID number encapsulated behind attribute 26.
Attribute	The ASCII string name of the attribute.
Description	Description of the attribute.

Table 111: Vendor-Specific RADIUS IETF Attributes

Number	Vendor-Specific Company Code	Sub-Type Number	Attribute	Description
MS-CHAP Attributes				
26	311	1	MSCHAP-Response	Contains the response value provided by a PPP MS-CHAP user in response to the challenge. It is only used in Access-Request packets. This attribute is identical to the PPP CHAP Identifier. (RFC 2548)
26	311	11	MSCHAP-Challenge	Contains the challenge sent by a network access server to an MS-CHAP user. It can be used in both Access-Request and Access-Challenge packets. (RFC 2548)
VPDN Attributes				

Number	Vendor-Specific Company Code	Sub-Type Number	Attribute	Description
26	9	1	l2tp-cm-local-window-size	Specifies the maximum receive window size for L2TP control messages. This value is advertised to the peer during tunnel establishment.
26	9	1	l2tp-drop-out-of-order	Respects sequence numbers on data packets by dropping those that are received out of order. This does not ensure that sequence numbers will be sent on data packets, just how to handle them if they are received.
26	9	1	l2tp-hello-interval	Specifies the number of seconds for the hello keepalive interval. Hello packets are sent when no data has been sent on a tunnel for the number of seconds configured here.
26	9	1	l2tp-hidden-avp	When enabled, sensitive AVPs in L2TP control messages are scrambled or hidden.
26	9	1	l2tp-nosession-timeout	Specifies the number of seconds that a tunnel will stay active with no sessions before timing out and shutting down.
26	9	1	tunnel-tos-reflect	Copies the IP ToS field from the IP header of each payload packet to the IP header of the tunnel packet for packets entering the tunnel at the LNS.
26	9	1	l2tp-tunnel-authen	If this attribute is set, it performs L2TP tunnel authentication.
26	9	1	l2tp-tunnel-password	Shared secret used for L2TP tunnel authentication and AVP hiding.

Number	Vendor-Specific Company Code	Sub-Type Number	Attribute	Description
26	9	1	l2tp-udp-checksum	This is an authorization attribute and defines whether L2TP should perform UDP checksums for data packets. Valid values are “yes” and “no.” The default is no.
Store and Forward Fax Attributes				
26	9	3	Fax-Account-Id-Origin	Indicates the account ID origin as defined by system administrator for the mmoip aaa receive-id or the mmoip aaa send-id commands.
26	9	4	Fax-Msg-Id=	Indicates a unique fax message identification number assigned by Store and Forward Fax.
26	9	5	Fax-Pages	Indicates the number of pages transmitted or received during this fax session. This page count includes cover pages.
26	9	6	Fax-Coverpage-Flag	Indicates whether or not a cover page was generated by the off-ramp gateway for this fax session. True indicates that a cover page was generated; false means that a cover page was not generated.
26	9	7	Fax-Modem-Time	Indicates the amount of time in seconds the modem sent fax data (x) and the amount of time in seconds of the total fax session (y), which includes both fax-mail and PSTN time, in the form x/y. For example, 10/15 means that the transfer time took 10 seconds, and the total fax session took 15 seconds.

Number	Vendor-Specific Company Code	Sub-Type Number	Attribute	Description
26	9	8	Fax-Connect-Speed	Indicates the modem speed at which this fax-mail was initially transmitted or received. Possible values are 1200, 4800, 9600, and 14400.
26	9	9	Fax-Recipient-Count	Indicates the number of recipients for this fax transmission. Until e-mail servers support Session mode, the number should be 1.
26	9	10	Fax-Process-Abort-Flag	Indicates that the fax session was cancelled or successful. True means that the session was cancelled; false means that the session was successful.
26	9	11	Fax-Dsn-Address	Indicates the address to which DSNs will be sent.
26	9	12	Fax-Dsn-Flag	Indicates whether or not DSN has been enabled. True indicates that DSN has been enabled; false means that DSN has not been enabled.
26	9	13	Fax-Mdn-Address	Indicates the address to which MDNs will be sent.
26	9	14	Fax-Mdn-Flag	Indicates whether or not message delivery notification (MDN) has been enabled. True indicates that MDN had been enabled; false means that MDN had not been enabled.
26	9	15	Fax-Auth-Status	Indicates whether or not authentication for this fax session was successful. Possible values for this field are success, failed, bypassed, or unknown.

Number	Vendor-Specific Company Code	Sub-Type Number	Attribute	Description
26	9	16	Email-Server-Address	Indicates the IP address of the e-mail server handling the on-ramp fax-mail message.
26	9	17	Email-Server-Ack-Flag	Indicates that the on-ramp gateway has received a positive acknowledgment from the e-mail server accepting the fax-mail message.
26	9	18	Gateway-Id	Indicates the name of the gateway that processed the fax session. The name appears in the following format: hostname.domain-name.
26	9	19	Call-Type	Describes the type of fax activity: fax receive or fax send.
26	9	20	Port-Used	Indicates the slot/port number of the Cisco AS5300 used to either transmit or receive this fax-mail.
26	9	21	Abort-Cause	If the fax session cancels, indicates the system component that signaled the cancel operation. Examples of system components that could trigger a cancel operation are FAP (Fax Application Process), TIFF (the TIFF reader or the TIFF writer), fax-mail client, fax-mail server, ESMTP client, or ESMTP server.
H323 Attributes				
26	9	23	Remote-Gateway-ID (h323-remote-address)	Indicates the IP address of the remote gateway.

Number	Vendor-Specific Company Code	Sub-Type Number	Attribute	Description
26	9	24	Connection-ID (h323-conf-id)	Identifies the conference ID.
26	9	25	Setup-Time (h323-setup-time)	Indicates the setup time for this connection in Coordinated Universal Time (UTC) formerly known as Greenwich Mean Time (GMT) and Zulu time.
26	9	26	Call-Origin (h323-call-origin)	Indicates the origin of the call relative to the gateway. Possible values are originating and terminating (answer).
26	9	27	Call-Type (h323-call-type)	Indicates call leg type. Possible values are telephony and VoIP .
26	9	28	Connect-Time (h323-connect-time)	Indicates the connection time for this call leg in UTC.
26	9	29	Disconnect-Time (h323-disconnect-time)	Indicates the time this call leg was disconnected in UTC.
26	9	30	Disconnect-Cause (h323-disconnect-cause)	Specifies the reason a connection was taken offline per Q.931 specification.
26	9	31	Voice-Quality (h323-voice-quality)	Specifies the impairment factor (ICPIF) affecting voice quality for a call.
26	9	33	Gateway-ID (h323-gw-id)	Indicates the name of the underlying gateway.
Large Scale Dialout Attributes				
26	9	1	callback-dialstring	Defines a dialing string to be used for callback.
26	9	1	data-service	No description available.
26	9	1	dial-number	Defines the number to dial.

Number	Vendor-Specific Company Code	Sub-Type Number	Attribute	Description
26	9	1	force-56	Determines whether the network access server uses only the 56 K portion of a channel, even when all 64 K appear to be available.
26	9	1	map-class	Allows the user profile to reference information configured in a map class of the same name on the network access server that dials out.
26	9	1	send-auth	Defines the protocol to use (PAP or CHAP) for username-password authentication following CLID authentication.

Number	Vendor-Specific Company Code	Sub-Type Number	Attribute	Description
26	9	1	send-name	<p>PPP name authentication. To apply for PAP, do not configure the ppp pap sent-name password command on the interface. For PAP, “preauth:send-name” and “preauth:send-secret” will be used as the PAP username and PAP password for outbound authentication. For CHAP, “preauth:send-name” will be used not only for outbound authentication, but also for inbound authentication. For a CHAP inbound case, the NAS will use the name defined in “preauth:send-name” in the challenge packet to the caller box.</p> <p>Note The send-name attribute has changed over time: Initially, it performed the functions now provided by both the send-name and remote-name attributes. Because the remote-name attribute has been added, the send-name attribute is restricted to its current behavior.</p>

Number	Vendor-Specific Company Code	Sub-Type Number	Attribute	Description
26	9	1	send-secret	PPP password authentication. The vendor-specific attributes (VSAs) "preauth:send-name" and "preauth:send-secret" will be used as the PAP username and PAP password for outbound authentication. For a CHAP outbound case, both "preauth:send-name" and "preauth:send-secret" will be used in the response packet.
26	9	1	remote-name	Provides the name of the remote host for use in large-scale dial-out. Dialer checks that the large-scale dial-out remote name matches the authenticated name, to protect against accidental user RADIUS misconfiguration. (For example, dialing a valid phone number but connecting to the wrong device.)
Miscellaneous Attributes				

Number	Vendor-Specific Company Code	Sub-Type Number	Attribute	Description
26	9	2	Cisco-NAS-Port	<p>Specifies additional vendor specific attribute (VSA) information for NAS-Port accounting. To specify additional NAS-Port information in the form an Attribute-Value Pair (AVPair) string, use the radius-server vsa send global configuration command.</p> <p>Note This VSA is typically used in Accounting, but may also be used in Authentication (Access-Request) packets.</p>
26	9	1	min-links	Sets the minimum number of links for MLP.
26	9	1	proxyacl#<n>	Allows users to configure the downloadable user profiles (dynamic ACLs) by using the authentication proxy feature so that users can have the configured authorization to permit traffic going through the configured interfaces.

Number	Vendor-Specific Company Code	Sub-Type Number	Attribute	Description
26	9	1	spi	Carries the authentication information needed by the home agent to authenticate a mobile node during registration. The information is in the same syntax as the ip mobile secure host <addr> configuration command. Basically it contains the rest of the configuration command that follows that string, verbatim. It provides the Security Parameter Index (SPI), key, authentication algorithm, authentication mode, and replay protection timestamp range.

RADIUS Disconnect-Cause Attribute Values

Disconnect-cause attribute values specify the reason a connection was taken offline. The attribute values are sent in Accounting request packets. These values are sent at the end of a session, even if the session fails to be authenticated. If the session is not authenticated, the attribute can cause stop records to be generated without first generating start records.

The table below lists the cause codes, values, and descriptions for the Disconnect-Cause (195) attribute.



Note The Disconnect-Cause is incremented by 1000 when it is used in RADIUS AVPairs; for example, disc-cause 4 becomes 1004.

Table 112: Disconnect-Cause Attribute Values

Cause Code	Value	Description
0	No-Reason	No reason is given for the disconnect.
1	No-Disconnect	The event was not disconnected.
2	Unknown	Reason unknown.
3	Call-Disconnect	The call has been disconnected.
4	CLID-Authentication-Failure	Failure to authenticate number of the calling-party.

Cause Code	Value	Description
9	No-Modem-Available	A modem is not available to connect the call.
10	No-Carrier	No carrier detected. Note Codes 10, 11, and 12 can be sent if there is a disconnection during initial modem connection.
11	Lost-Carrier	Loss of carrier.
12	No-Detected-Result-Codes	Failure to detect modem result codes.
20	User-Ends-Session	User terminates a session. Note Codes 20, 22, 23, 24, 25, 26, 27, and 28 apply to EXEC sessions.
21	Idle-Timeout	Timeout waiting for user input. Codes 21, 100, 101, 102, and 120 apply to all session types.
22	Exit-Telnet-Session	Disconnect due to exiting Telnet session.
23	No-Remote-IP-Addr	Could not switch to SLIP/PPP; the remote end has no IP address.
24	Exit-Raw-TCP	Disconnect due to exiting raw TCP.
25	Password-Fail	Bad passwords.
26	Raw-TCP-Disabled	Raw TCP disabled.
27	Control-C-Detected	Control-C detected.
28	EXEC-Process-Destroyed	EXEC process destroyed.
29	Close-Virtual-Connection	User closes a virtual connection.
30	End-Virtual-Connection	Virtual connection has ended.
31	Exit-Rlogin	User exits Rlogin.
32	Invalid-Rlogin-Option	Invalid Rlogin option selected.
33	Insufficient-Resources	Insufficient resources.
40	Timeout-PPP-LCP	PPP LCP negotiation timed out. Note Codes 40 through 49 apply to PPP sessions.
41	Failed-PPP-LCP-Negotiation	PPP LCP negotiation failed.
42	Failed-PPP-PAP-Auth-Fail	PPP PAP authentication failed.
43	Failed-PPP-CHAP-Auth	PPP CHAP authentication failed.
44	Failed-PPP-Remote-Auth	PPP remote authentication failed.

Cause Code	Value	Description
45	PPP-Remote-Terminate	PPP received a Terminate Request from remote end.
46	PPP-Closed-Event	Upper layer requested that the session be closed.
47	NCP-Closed-PPP	PPP session closed because there were no NCPs open.
48	MP-Error-PPP	PPP session closed because of an MP error.
49	PPP-Maximum-Channels	PPP session closed because maximum channels were reached.
50	Tables-Full	Disconnect due to full terminal server tables.
51	Resources-Full	Disconnect due to full internal resources.
52	Invalid-IP-Address	IP address is not valid for Telnet host.
53	Bad-Hostname	Hostname cannot be validated.
54	Bad-Port	Port number is invalid or missing.
60	Reset-TCP	TCP connection has been reset. Note Codes 60 through 67 apply to Telnet or raw TCP sessions.
61	TCP-Connection-Refused	TCP connection has been refused by the host.
62	Timeout-TCP	TCP connection has timed out.
63	Foreign-Host-Close-TCP	TCP connection has been closed.
64	TCP-Network-Unreachable	TCP network is unreachable.
65	TCP-Host-Unreachable	TCP host is unreachable.
66	TCP-Network-Admin Unreachable	TCP network is unreachable for administrative reasons.
67	TCP-Port-Unreachable	TCP port in unreachable.
100	Session-Timeout	Session timed out.
101	Session-Failed-Security	Session failed for security reasons.
102	Session-End-Callback	Session terminated due to callback.
120	Invalid-Protocol	Call refused because the detected protocol is disabled.
150	RADIUS-Disconnect	Disconnected by RADIUS request.
151	Local-Admin-Disconnect	Administrative disconnect.
152	SNMP-Disconnect	Disconnected by SNMP request.
160	V110-Retries	Allowed V.110 retries have been exceeded.
170	PPP-Authentication-Timeout	PPP authentication timed out.

Cause Code	Value	Description
180	Local-Hangup	Disconnected by local hangup.
185	Remote-Hangup	Disconnected by remote end hangup.
190	T1-Quiesced	Disconnected because T1 line was quiesced.
195	Call-Duration	Disconnected because the maximum duration of the call was exceeded.
600	VPN-User-Disconnect	Call disconnected by client (through PPP). Code is sent if the LNS receives a PPP terminate request from the client.
601	VPN-Carrier-Loss	Loss of carrier. This can be the result of a physical line going dead. Code is sent when a client is unable to dial out using a dialer.
602	VPN-No-Resources	No resources available to handle the call. Code is sent when the client is unable to allocate memory (running low on memory).
603	VPN-Bad-Control-Packet	Bad L2TP or L2F control packets. This code is sent when an invalid control packet, such as missing mandatory Attribute-Value pairs (AVP), from the peer is received. When using L2TP, the code will be sent after six retransmits; when using L2F, the number of retransmits is user configurable. Note VPN-Tunnel-Shut will be sent if there are active sessions in the tunnel.
604	VPN-Admin-Disconnect	Administrative disconnect. This can be the result of a VPN soft shutdown, which is when a client reaches maximum session limit or exceeds maximum hopcount. Code is sent when a tunnel is brought down by issuing the clear vpdn tunnel command.
605	VPN-Tunnel-Shut	Tunnel teardown or tunnel setup has failed. Code is sent when there are active sessions in a tunnel and the tunnel goes down. Note This code is not sent when tunnel authentication fails.
606	VPN-Local-Disconnect	Call is disconnected by LNS PPP module. Code is sent when the LNS sends a PPP terminate request to the client. It indicates a normal PPP disconnection initiated by the LNS.
607	VPN-Session-Limit	VPN soft shutdown is enabled. Code is sent when a call has been refused due to any of the soft shutdown restrictions previously mentioned.
608	VPN-Call-Redirect	VPN call redirect is enabled.

RADIUS Progress Codes

The RADIUS Progress Codes feature adds additional progress codes to RADIUS attribute 196 (Ascend-Connect-Progress), which indicates a connection state before a call is disconnected through progress codes.

Attribute 196 is sent in network, exec, and resource accounting “start” and “stop” records. This attribute can facilitate call failure debugging because each progress code identifies accounting information relevant to the connection state of a call. The attribute is activated by default; when an accounting “start” or “stop” accounting record is requested, authentication, authorization, and accounting (AAA) adds attribute 196 into the record as part of the standard attribute list. Attribute 196 is valuable because the progress codes, which are sent in accounting “start” and “stop” records, facilitate the debugging of call failures.



Note In accounting “start” records, attribute 196 does not have a value.

Table 113: Newly Supported Progress Codes for Attribute 196

Code	Description
10	Modem allocation and negotiation is complete; the call is up.
30	The modem is up.
33	The modem is waiting for result codes.
41	The max TNT is establishing the TCP connection by setting up a TCP clear call.
60	Link control protocol (LCP) is the open state with PPP and IP Control Protocol (IPCP) negotiation; the LAN session is up.
65	PPP negotiation occurs and, initially, the LCP negotiation occurs; LCP is in the open state.
67	After PPP negotiation with LCP in the open state occurs, IPCP negotiation begins.



Note Progress codes 33, 30, and 67 are generated and seen through debugs on the NAS; all other codes are generated and seen through debugs and the accounting record on the RADIUS server.

Vendor-Proprietary RADIUS Server Communication

Although an IETF draft standard for RADIUS specifies a method for communicating vendor-proprietary information between the switch and the RADIUS server, some vendors have extended the RADIUS attribute set in a unique way. Cisco IOS software supports a subset of vendor-proprietary RADIUS attributes.

As mentioned earlier, to configure RADIUS (whether vendor-proprietary or IETF draft-compliant), you must specify the host running the RADIUS server daemon and the secret text string it shares with the switch. You specify the RADIUS host and secret text string by using the **radius server** global configuration commands.

Enhanced Test Command

The Enhanced Test Command feature allows a named user profile to be created with calling line ID (CLID) or dialed number identification service (DNIS) attribute values. The CLID or DNIS attribute values can be associated with the RADIUS record that is sent with the user profile so that the RADIUS server can access CLID or DNIS attribute information for all incoming calls.

How to Configure RADIUS

Identifying the RADIUS Server Host

To apply these settings globally to all RADIUS servers communicating with the Device, use the three unique global configuration commands: **radius-server timeout**, **radius-server retransmit**, and **radius-server key**.

You can configure the Device to use AAA server groups to group existing server hosts for authentication. For more information, see Related Topics below.

You also need to configure some settings on the RADIUS server. These settings include the IP address of the Device and the key string to be shared by both the server and the Device. For more information, see the RADIUS server documentation.

Follow these steps to configure per-server RADIUS server communication.

Before you begin

If you configure both global and per-server functions (timeout, retransmission, and key commands) on the device, the per-server timer, retransmission, and key value commands override global timer, retransmission, and key value commands. For information on configuring these settings on all RADIUS servers, see Related Topics below.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **radius server** *name*
4. **address** {*ipv4* | *ipv6*} *ip address* {**auth-port** *port number* | **acct-port** *port number*}
5. **key** *string*
6. **retransmit** *value*
7. **timeout** *seconds*
8. **end**
9. **show running-config**
10. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example:	Enables privileged EXEC mode. • Enter your password if prompted.

	Command or Action	Purpose
	Device> enable	
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	radius server <i>name</i> Example: Device(config)# radius server ISE	Specifies the name of the RADIUS server configuration for Protected Access Credential (PAC) provisioning and enters RADIUS server configuration mode. The device also supports RADIUS for IPv6.
Step 4	address { ipv4 ipv6 } <i>ip address</i> { auth-port <i>port number</i> acct-port <i>port number</i> } Example: Device(config-radius-server)# address ipv4 10.1.1.1 auth-port 1645 acct-port 1646	(Optional) Specifies the RADIUS server parameters. For auth-port <i>port-number</i> , specify the UDP destination port for authentication requests. The default is 1645. The range is 0 to 65536. For acct-port <i>port-number</i> , specify the UDP destination port for authentication requests. The default is 1646.
Step 5	key <i>string</i> Example: Device(config-radius-server)# key cisco123	(Optional) For key <i>string</i> , specify the authentication and encryption key used between the Device and the RADIUS daemon running on the RADIUS server. Note The key is a text string that must match the encryption key used on the RADIUS server. Always configure the key as the last item in the radius server command. Leading spaces are ignored, but spaces within and at the end of the key are used. If you use spaces in your key, do not enclose the key in quotation marks unless the quotation marks are part of the key.
Step 6	retransmit <i>value</i> Example: Device(config-radius-server)# retransmit 10	(Optional) Specifies the number of times a RADIUS request is resent when the server is not responding or responding slowly. The range is 1 to 100. This setting overrides the radius-server retransmit global configuration command setting.
Step 7	timeout <i>seconds</i> Example: Device(config-radius-server)# timeout 60	(Optional) Specifies the time interval that the Device waits for the RADIUS server to reply before sending a request again. The range is 1 to 1000. This setting overrides the radius-server timeout global configuration command setting.

	Command or Action	Purpose
Step 8	end Example: Device(config)# end	Returns to privileged EXEC mode.
Step 9	show running-config Example: Device# show running-config	Verifies your entries.
Step 10	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Configuring Settings for All RADIUS Servers

Beginning in privileged EXEC mode, follow these steps to configure settings for all RADIUS servers:

SUMMARY STEPS

1. **configure terminal**
2. **radius-server key *string***
3. **radius-server retransmit *retries***
4. **radius-server timeout *seconds***
5. **radius-server deadtime *minutes***
6. **end**
7. **show running-config**
8. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 2	radius-server key <i>string</i> Example:	Specifies the shared secret text string used between the switch and all RADIUS servers.

	Command or Action	Purpose
	<pre>Device(config)# radius-server key your_server_key</pre> <pre>Device(config)# key your_server_key</pre>	<p>Note The key is a text string that must match the encryption key used on the RADIUS server. Leading spaces are ignored, but spaces within and at the end of the key are used. If you use spaces in your key, do not enclose the key in quotation marks unless the quotation marks are part of the key.</p>
Step 3	<p>radius-server retransmit <i>retries</i></p> <p>Example:</p> <pre>Device(config)# radius-server retransmit 5</pre>	Specifies the number of times the switch sends each RADIUS request to the server before giving up. The default is 3; the range 1 to 1000.
Step 4	<p>radius-server timeout <i>seconds</i></p> <p>Example:</p> <pre>Device(config)# radius-server timeout 3</pre>	Specifies the number of seconds a switch waits for a reply to a RADIUS request before resending the request. The default is 5 seconds; the range is 1 to 1000.
Step 5	<p>radius-server deadtime <i>minutes</i></p> <p>Example:</p> <pre>Device(config)# radius-server deadtime 0</pre>	When a RADIUS server is not responding to authentication requests, this command specifies a time to stop the request on that server. This avoids the wait for the request to timeout before trying the next configured server. The default is 0; the range is 1 to 1440 minutes.
Step 6	<p>end</p> <p>Example:</p> <pre>Device(config)# end</pre>	Returns to privileged EXEC mode.
Step 7	<p>show running-config</p> <p>Example:</p> <pre>Device# show running-config</pre>	Verifies your entries.
Step 8	<p>copy running-config startup-config</p> <p>Example:</p> <pre>Device# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Configuring RADIUS Login Authentication

Follow these steps to configure RADIUS login authentication:

Before you begin

To secure the device for HTTP access by using AAA methods, you must configure the device with the **ip http authentication aaa** global configuration command. Configuring AAA authentication does not secure the device for HTTP access by using AAA methods.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **aaa new-model**
4. **aaa authentication login {default | list-name} method1 [method2...]**
5. **line [console | tty | vty] line-number [ending-line-number]**
6. **login authentication {default | list-name}**
7. **end**
8. **show running-config**
9. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	aaa new-model Example: Device(config)# aaa new-model	Enables AAA.
Step 4	aaa authentication login {default list-name} method1 [method2...] Example: Device(config)# aaa authentication login default local	Creates a login authentication method list. <ul style="list-style-type: none"> • To create a default list that is used when a named list is <i>not</i> specified in the login authentication command, use the default keyword followed by the methods that are to be used in default situations. The default method list is automatically applied to all ports. • For <i>list-name</i>, specify a character string to name the list you are creating.

	Command or Action	Purpose
		<ul style="list-style-type: none"> For <i>method1...</i>, specify the actual method the authentication algorithm tries. The additional methods of authentication are used only if the previous method returns an error, not if it fails. <p>Select one of these methods:</p> <ul style="list-style-type: none"> <i>enable</i>—Use the enable password for authentication. Before you can use this authentication method, you must define an enable password by using the enable password global configuration command. <i>group radius</i>—Use RADIUS authentication. Before you can use this authentication method, you must configure the RADIUS server. <i>line</i>—Use the line password for authentication. Before you can use this authentication method, you must define a line password. Use the password password line configuration command. <i>local</i>—Use the local username database for authentication. You must enter username information in the database. Use the username name password global configuration command. <i>local-case</i>—Use a case-sensitive local username database for authentication. You must enter username information in the database by using the username password global configuration command. <i>none</i>—Do not use any authentication for login.
Step 5	line [console tty vty] <i>line-number</i> [<i>ending-line-number</i>] Example: <pre>Device(config)# line 1 4</pre>	Enters line configuration mode, and configure the lines to which you want to apply the authentication list.
Step 6	login authentication {default <i>list-name</i> } Example: <pre>Device(config)# login authentication default</pre>	Applies the authentication list to a line or set of lines. <ul style="list-style-type: none"> If you specify default, use the default list created with the aaa authentication login command. For <i>list-name</i>, specify the list created with the aaa authentication login command.
Step 7	end Example:	Returns to privileged EXEC mode.

	Command or Action	Purpose
	Device(config)# end	
Step 8	show running-config Example: Device# show running-config	Verifies your entries.
Step 9	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Defining AAA Server Groups

You use the **server** group server configuration command to associate a particular server with a defined group server. You can either identify the server by its IP address or identify multiple host instances or entries by using the optional **auth-port** and **acct-port** keywords.

Follow these steps to define AAA server groups:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **radius server** *name*
4. **address** {**ipv4** | **ipv6**} {*ip-address* | *hostname*} **auth-port** *port-number* **acct-port** *port-number*
5. **key** *string*
6. **end**
7. **show running-config**
8. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example:	Enters global configuration mode.

	Command or Action	Purpose
	Device# <code>configure terminal</code>	
Step 3	radius server <i>name</i> Example: Device(config)# <code>radius server ISE</code>	Specifies the name of the RADIUS server configuration for Protected Access Credential (PAC) provisioning and enters RADIUS server configuration mode. The device also supports RADIUS for IPv6.
Step 4	address { ipv4 ipv6 } { <i>ip-address</i> <i>hostname</i> } auth-port <i>port-number</i> acct-port <i>port-number</i> Example: Device(config-radius-server)# <code>address ipv4 10.1.1.1 auth-port 1645 acct-port 1646</code>	Configures the IPv4 address for the RADIUS server accounting and authentication parameters.
Step 5	key <i>string</i> Example: Device(config-radius-server)# <code>key cisco123</code>	Specifies the authentication and encryption key for all RADIUS communications between the device and the RADIUS server.
Step 6	end Example: Device(config-radius-server)# <code>end</code>	Exits RADIUS server configuration mode and returns to privileged EXEC mode.
Step 7	show running-config Example: Device# <code>show running-config</code>	Verifies your entries.
Step 8	copy running-config startup-config Example: Device# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Configuring RADIUS Authorization for User Privileged Access and Network Services



Note Authorization is bypassed for authenticated users who log in through the CLI even if authorization has been configured.

Follow these steps to configure RADIUS authorization for user privileged access and network services:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **aaa authorization network radius**
4. **aaa authorization exec radius**
5. **end**
6. **show running-config**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	aaa authorization network radius Example: Device(config)# aaa authorization network radius	Configures the device for user RADIUS authorization for all network-related service requests.
Step 4	aaa authorization exec radius Example: Device(config)# aaa authorization exec radius	Configures the device for user RADIUS authorization if the user has privileged EXEC access. The exec keyword might return user profile information (such as autocommand information).

	Command or Action	Purpose
Step 5	end Example: Device(config)# end	Returns to privileged EXEC mode.
Step 6	show running-config Example: Device# show running-config	Verifies your entries.
Step 7	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

What to do next

You can use the **aaa authorization** global configuration command with the **radius** keyword to set parameters that restrict a user's network access to privileged EXEC mode.

The **aaa authorization exec radius local** command sets these authorization parameters:

- Use RADIUS for privileged EXEC access authorization if authentication was performed by using RADIUS.
- Use the local database if authentication was not performed by using RADIUS.

Starting RADIUS Accounting

Follow these steps to start RADIUS accounting:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **aaa accounting network start-stop radius**
4. **aaa accounting exec start-stop radius**
5. **end**
6. **show running-config**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable	Enables privileged EXEC mode.

	Command or Action	Purpose
	Example: Device> <code>enable</code>	<ul style="list-style-type: none"> Enter your password if prompted.
Step 2	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode.
Step 3	aaa accounting network start-stop radius Example: Device(config)# <code>aaa accounting network start-stop radius</code>	Enables RADIUS accounting for all network-related service requests.
Step 4	aaa accounting exec start-stop radius Example: Device(config)# <code>aaa accounting exec start-stop radius</code>	Enables RADIUS accounting to send a start-record accounting notice at the beginning of a privileged EXEC process and a stop-record at the end.
Step 5	end Example: Device(config)# <code>end</code>	Returns to privileged EXEC mode.
Step 6	show running-config Example: Device# <code>show running-config</code>	Verifies your entries.
Step 7	copy running-config startup-config Example: Device# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Verifying Attribute 196

No configuration is required to configure RADIUS Progress Codes. To verify attribute 196 in accounting “start” and “stop” records, perform the following steps.

SUMMARY STEPS

1. **enable**
2. **debug aaa accounting**
3. **show radius statistics**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	debug aaa accounting Example: Device# debug aaa accounting	Displays information on accountable events as they occur.
Step 3	show radius statistics Example: Device# debug aaa authorization	Displays the RADIUS statistics for accounting and authentication packets.

Configuring the Device to Use Vendor-Specific RADIUS Attributes

Follow these steps to configure the device to use vendor-specific RADIUS attributes:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **radius-server vsa send [accounting | authentication]**
4. **end**
5. **show running-config**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.

	Command or Action	Purpose
Step 2	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode.
Step 3	radius-server vsa send [accounting authentication] Example: Device(config)# <code>radius-server vsa send accounting</code>	Enables the device to recognize and use VSAs as defined by RADIUS IETF attribute 26. <ul style="list-style-type: none"> • (Optional) Use the accounting keyword to limit the set of recognized vendor-specific attributes to only accounting attributes. • (Optional) Use the authentication keyword to limit the set of recognized vendor-specific attributes to only authentication attributes. If you enter this command without keywords, both accounting and authentication vendor-specific attributes are used.
Step 4	end Example: Device(config)# <code>end</code>	Returns to privileged EXEC mode.
Step 5	show running-config Example: Device# <code>show running-config</code>	Verifies your entries.
Step 6	copy running-config startup-config Example: Device# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Configuring the Device for Vendor-Proprietary RADIUS Server Communication

Follow these steps to configure the device to use vendor-proprietary RADIUS server communication:

SUMMARY STEPS

1. `enable`
2. `configure terminal`
3. `radius-server host {hostname | ip-address} non-standard`
4. `radius-server key string`

5. `end`
6. `show running-config`
7. `copy running-config startup-config`

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p>enable</p> <p>Example:</p> <pre>Device> enable</pre>	<p>Enables privileged EXEC mode.</p> <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	<p>configure terminal</p> <p>Example:</p> <pre>Device# configure terminal</pre>	<p>Enters global configuration mode.</p>
Step 3	<p>radius-server host {hostname ip-address} non-standard</p> <p>Example:</p> <pre>Device(config)# radius-server host 172.20.30.15 non-standard</pre>	<p>Specifies the IP address or hostname of the remote RADIUS server host and identifies that it is using a vendor-proprietary implementation of RADIUS.</p>
Step 4	<p>radius-server key string</p> <p>Example:</p> <pre>Device(config)# radius-server key rad124</pre>	<p>Specifies the shared secret text string used between the device and the vendor-proprietary RADIUS server. The device and the RADIUS server use this text string to encrypt passwords and exchange responses.</p> <p>Note The key is a text string that must match the encryption key used on the RADIUS server. Leading spaces are ignored, but spaces within and at the end of the key are used. If you use spaces in your key, do not enclose the key in quotation marks unless the quotation marks are part of the key.</p>
Step 5	<p>end</p> <p>Example:</p> <pre>Device(config)# end</pre>	<p>Returns to privileged EXEC mode.</p>
Step 6	<p>show running-config</p> <p>Example:</p> <pre>Device# show running-config</pre>	<p>Verifies your entries.</p>

	Command or Action	Purpose
Step 7	copy running-config startup-config Example: Device# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Configuring a User Profile and Associating it with the RADIUS Record

This section describes how to create a named user profile with CLID or DNIS attribute values and associate it with the RADIUS record.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **aaa user profile** *profile-name*
4. **aaa attribute** {dnis | clid}
5. **exit**
6. **test aaa group** {group-name | radius} username password new-code [profile profile-name]

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	aaa user profile <i>profile-name</i> Example: Device(config)# aaa user profile profilename1	Creates a user profile.
Step 4	aaa attribute {dnis clid} Example: Device# configure terminal	Adds DNIS or CLID attribute values to the user profile and enters AAA-user configuration mode.
Step 5	exit	Exit Global Configuration mode.

	Command or Action	Purpose
Step 6	test aaa group {group-name radius} username password new-code [profile profile-name] Example: Device# test aaa group radius secret new-code profile profilename1	Associates a DNIS or CLID named user profile with the record sent to the RADIUS server. Note The <i>profile-name</i> must match the profile-name specified in the aaa user profile command.

Verifying the Enhanced Test Command Configuration

To verify the Enhanced Test Command configuration, use the following commands in privileged EXEC mode:

Command	Purpose
Device# debug radius	Displays information associated with RADIUS.
Device# more system:running-config	Displays the contents of the current running configuration file. (Note that the more system:running-config command has replaced the show running-config command.)

Configuration Examples for RADIUS

Examples: Identifying the RADIUS Server Host

This example shows how to configure one RADIUS server to be used for authentication and another to be used for accounting:

```
Device(config)# radius-server host 172.29.36.49 auth-port 1612 key rad1
Device(config)# radius-server host 172.20.36.50 acct-port 1618 key rad2
```

This example shows how to configure *host1* as the RADIUS server and to use the default ports for both authentication and accounting:

```
Device(config)# radius-server host host1
```

Example: Using Two Different RADIUS Group Servers

In this example, the switch is configured to recognize two different RADIUS group servers (*group1* and *group2*). Group1 has two different host entries on the same RADIUS server configured for the same services. The second host entry acts as a fail-over backup to the first entry.

```
Device(config)# radius-server host 172.20.0.1 auth-port 1000 acct-port 1001
Device(config)# radius-server host 172.10.0.1 auth-port 1645 acct-port 1646
Device(config)# aaa new-model
```

```

Device(config)# aaa group server radius group1
Device(config-sg-radius)# server 172.20.0.1 auth-port 1000 acct-port 1001
Device(config-sg-radius)# exit
Device(config)# aaa group server radius group2
Device(config-sg-radius)# server 172.20.0.1 auth-port 2000 acct-port 2001
Device(config-sg-radius)# exit

```

Examples: AAA Server Groups

The following example shows how to create server group radgroup1 with three different RADIUS server members, each using the default authentication port (1645) and accounting port (1646):

```

aaa group server radius radgroup1
server 172.16.1.11
server 172.17.1.21
server 172.18.1.31

```

The following example shows how to create server group radgroup2 with three RADIUS server members, each with the same IP address but with unique authentication and accounting ports:

```

aaa group server radius radgroup2
server 172.16.1.1 auth-port 1000 acct-port 1001
server 172.16.1.1 auth-port 2000 acct-port 2001
server 172.16.1.1 auth-port 3000 acct-port 3001

```

Troubleshooting Tips for RADIUS Progress Codes

The following example is a sample debug output from the **debug ppp negotiation** command. This debug output is used to verify that accounting “stop” records have been generated and that attribute 196 (Ascend-Connect-Progress) has a value of 65.

```

Tue Aug 7 06:21:03 2001
NAS-IP-Address = 10.0.58.62
NAS-Port = 20018
Vendor-Specific = ""
NAS-Port-Type = ISDN
User-Name = "peer_16a"
Called-Station-Id = "5213124"
Calling-Station-Id = "5212175"
Acct-Status-Type = Stop
Acct-Authentic = RADIUS
Service-Type = Framed-User
Acct-Session-Id = "00000014"
Framed-Protocol = PPP
Framed-IP-Address = 172.16.0.2
Acct-Input-Octets = 3180
Acct-Output-Octets = 3186
Acct-Input-Packets = 40
Acct-Output-Packets = 40
Ascend-Connect-Pr = 65
Acct-Session-Time = 49
Acct-Delay-Time = 0
Timestamp = 997190463
Request-Authenticator = Unverified

```

Examples: Configuring the Switch to Use Vendor-Specific RADIUS Attributes

For example, this AV pair activates Cisco's *multiple named ip address pools* feature during IP authorization (during PPP IPCP address assignment):

```
cisco-avpair= "ip:addr-pool=first"
```

This example shows how to provide a user logging in from a switch with immediate access to privileged EXEC commands:

```
cisco-avpair= "shell:priv-lvl=15"
```

This example shows how to specify an authorized VLAN in the RADIUS server database:

```
cisco-avpair= "tunnel-type (#64)=VLAN(13) "
cisco-avpair= "tunnel-medium-type (#65)=802 media (6) "
cisco-avpair= "tunnel-private-group-id (#81)=vlanid"
```

This example shows how to apply an input ACL in ASCII format to an interface for the duration of this connection:

```
cisco-avpair= "ip:inacl#1=deny ip 10.10.10.10 0.0.255.255 20.20.20.20 255.255.0.0"
cisco-avpair= "ip:inacl#2=deny ip 10.10.10.10 0.0.255.255 any"
cisco-avpair= "mac:inacl#3=deny any any deernet-iv"
```

This example shows how to apply an output ACL in ASCII format to an interface for the duration of this connection:

```
cisco-avpair= "ip:outacl#2=deny ip 10.10.10.10 0.0.255.255 any"
```

Example: Configuring the Switch for Vendor-Proprietary RADIUS Server Communication

This example shows how to specify a vendor-proprietary RADIUS host and to use a secret key of *rad124* between the switch and the server:

```
Device(config)# radius-server host 172.20.30.15 nonstandard
Device(config)# radius-server key rad124
```

Example: User Profile Associated With the test aaa group Command

The following example shows how to configure the `dnis = dnisvalue` user profile "prfl1" and associate it with a **test aaa group** command. In this example, the **debug radius** command has been enabled and the output follows the configuration.

```
aaa user profile prfl1
  aaa attribute dnis
```

```

aaa attribute dnis dnisvalue
no aaa attribute clid
! Attribute not found.
aaa attribute clid clidvalue
no aaa attribute clid
exit
!
! Associate the dnis user profile with the test aaa group command.
test aaa group radius user1 pass new-code profile profl1
!
!
! debug radius output, which shows that the dnis value has been passed to the radius !
server.
*Dec 31 16:35:48: RADIUS: Sending packet for Unique id = 0
*Dec 31 16:35:48: RADIUS: Initial Transmit unknown id 8 172.22.71.21:1645, Access-Request,
len 68
*Dec 31 16:35:48: RADIUS: code=Access-Request id=08 len=0068
    authenticator=1E CA 13 F2 E2 81 57 4C - 02 EA AF 9D 30 D9 97 90
    T=User-Password[2]                L=12 V=*
    T=User-Name[1]                    L=07 V="test"
    T=Called-Station-Id[30]           L=0B V="dnisvalue"
    T=Service-Type[6]                L=06 V>Login
    T=NAS-IP-Address[4]              L=06 V=10.0.1.81
                                     [1]
*Dec 31 16:35:48: RADIUS: Received from id 8 172.22.71.21:1645, Access-Accept, len 38
*Dec 31 16:35:48: RADIUS: code=Access-Accept id=08 len=0038

```

Additional References for RADIUS

Related Documents

Related Topic	Document Title
Cisco security commands	<ul style="list-style-type: none"> • Cisco IOS Security Command Reference: Commands A to C • Cisco IOS Security Command Reference: Commands D to L • Cisco IOS Security Command Reference: Commands M to R • Cisco IOS Security Command Reference: Commands S to Z
IPv6 commands	Cisco IOS IPv6 Command Reference

Standards and RFCs

Standard/RFC	Title
RFC 5176	RADIUS Change of Authorization (CoA) extensions

Error Message Decoder

Description	Link
To help you research and resolve system error messages in this release, use the Error Message Decoder tool.	https://www.cisco.com/cgi-bin/Support/Errordecoder/index.cgi

MIBs

MIB	MIBs Link
All the supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
<p>The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies.</p> <p>To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds.</p> <p>Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.</p>	http://www.cisco.com/support

Feature Information for RADIUS

Release	Feature Information
Cisco IOS Release 15.0(2)EX1	This feature was introduced.
Cisco IOS 15.2(1)E	The RADIUS Progress Codes feature adds additional progress codes to RADIUS attribute 196 (Ascend-Connect-Progress), which indicates a connection state before a call is disconnected through progress codes.

Release	Feature Information
Cisco IOS 15.2(1)E	<p>The Enhanced Test Command feature allows a named user profile to be created with calling line ID (CLID) or Dialed Number Identification Service (DNIS) attribute values. The CLID or DNIS attribute values can be associated with the RADIUS record that is sent with the user profile so that the RADIUS server can access CLID or DNIS attribute information for all incoming calls.</p> <p>The following commands were introduced or modified: aaa attribute, aaa user profile, and test aaa group</p>



CHAPTER 52

RADIUS Server Load Balancing

The RADIUS Server Load Balancing feature distributes authentication, authorization, and accounting (AAA) authentication and accounting transactions across RADIUS servers in a server group. These servers can share the AAA transaction load and thereby respond faster to incoming requests.

This module describes the RADIUS Server Load Balancing feature.

- [Finding Feature Information, on page 1101](#)
- [Prerequisites for RADIUS Server Load Balancing, on page 1101](#)
- [Restrictions for RADIUS Server Load Balancing, on page 1102](#)
- [Information About RADIUS Server Load Balancing, on page 1102](#)
- [How to Configure RADIUS Server Load Balancing, on page 1104](#)
- [Configuration Examples for RADIUS Server Load Balancing, on page 1108](#)
- [Additional References for RADIUS Server Load Balancing, on page 1115](#)
- [Feature Information for RADIUS Server Load Balancing, on page 1115](#)

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to <http://www.cisco.com/go/cfn>. An account on Cisco.com is not required.

Prerequisites for RADIUS Server Load Balancing

- Authentication, authorization, and accounting (AAA) must be configured on the RADIUS server.
- AAA RADIUS server groups must be configured.
- RADIUS must be configured for functions such as authentication, accounting, or static route download.

Restrictions for RADIUS Server Load Balancing

- Incoming RADIUS requests, such as Packet of Disconnect (POD) requests, are not supported.
- Load balancing is not supported on proxy RADIUS servers and for private server groups.

Information About RADIUS Server Load Balancing

RADIUS Server Load Balancing Overview

Load balancing distributes batches of transactions to RADIUS servers within a server group. Load balancing assigns each batch of transactions to the server with the lowest number of outstanding transactions in its queue. The process of assigning a batch of transactions is as follows:

1. The first transaction is received for a new batch.
2. All server transaction queues are checked.
3. The server with the lowest number of outstanding transactions is identified.
4. The identified server is assigned the next batch of transactions.

The batch size is a user-configured parameter. Changes in the batch size may impact CPU load and network throughput. As batch size increases, CPU load decreases and network throughput increases. However, if a large batch size is used, all available server resources may not be fully utilized. As batch size decreases, CPU load increases and network throughput decreases.



Note There is no set number for large or small batch sizes. A batch with more than 50 transactions is considered large and a batch with fewer than 25 transactions is considered small.



Note If a server group contains ten or more servers, we recommend that you set a high batch size to reduce CPU load.

Transaction Load Balancing Across RADIUS Server Groups

You can configure load balancing either per-named RADIUS server group or for the global RADIUS server group. The load balancing server group must be referred to as “radius” in the authentication, authorization, and accounting (AAA) method lists. All public servers that are part of the RADIUS server group are then load balanced.

You can configure authentication and accounting to use the same RADIUS server or different servers. In some cases, the same server can be used for preauthentication, authentication, or accounting transactions for a session. The preferred server, which is an internal setting and is set as the default, informs AAA to use the same server for the start and stop record for a session regardless of the server cost. When using the preferred

server setting, ensure that the server that is used for the initial transaction (for example, authentication), the preferred server, is part of any other server group that is used for a subsequent transaction (for example, accounting).

The preferred server is not used if one of the following criteria is true:

- The **load-balance method least-outstanding ignore-preferred-server** command is used.
- The preferred server is dead.
- The preferred server is in quarantine.
- The want server flag has been set, overriding the preferred server setting.

The want server flag, an internal setting, is used when the same server must be used for all stages of a multistage transaction regardless of the server cost. If the want server is not available, the transaction fails.

You can use the **load-balance method least-outstanding ignore-preferred-server** command if you have either of the following configurations:

- Dedicated authentication server and a separate dedicated accounting server
- Network where you can track all call record statistics and call record details, including start and stop records and records that are stored on separate servers

If you have a configuration where authentication servers are a superset of accounting servers, the preferred server is not used.

RADIUS Server Status and Automated Testing

The RADIUS Server Load Balancing feature considers the server status when assigning batches. Transaction batches are sent only to live servers. We recommend that you test the status of all RADIUS load-balanced servers, including low usage servers (for example, backup servers).

Transactions are not sent to a server that is marked dead. A server is marked dead until its timer expires, at which time it moves to quarantine state. A server is in quarantine until it is verified alive by the RADIUS automated tester functionality.

To determine if a server is alive and available to process transactions, the RADIUS automated tester sends a request periodically to the server for a test user ID. If the server returns an Access-Reject message, the server is alive; otherwise the server is either dead or quarantined.

A transaction sent to an unresponsive server is failed over to the next available server before the unresponsive server is marked dead. We recommend that you use the retry reorder mode for failed transactions.

When using the RADIUS automated tester, verify that the authentication, authorization, and accounting (AAA) servers are responding to the test packets that are sent by the network access server (NAS). If the servers are not configured correctly, packets may be dropped and the server erroneously marked dead.



Caution

We recommend that you use a test user that is not defined on the RADIUS server for the RADIUS server automated testing to protect against security issues that may arise if the test user is not correctly configured.



Note Use the **test aaa group** command to check load-balancing transactions.

How to Configure RADIUS Server Load Balancing

Enabling Load Balancing for a Named RADIUS Server Group

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **radius-server host** *{hostname | ip-address}* [**test username name**] [**auth-port number**] [**ignore-auth-port**] [**acct-port number**] [**ignore-acct-port**] [**idle-time seconds**]
4. **aaa group server radius** *group-name*
5. **load-balance method** **least-outstanding** [**batch-size number**] [**ignore-preferred-server**]
6. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	radius-server host <i>{hostname ip-address}</i> [test username name] [auth-port number] [ignore-auth-port] [acct-port number] [ignore-acct-port] [idle-time seconds] Example: Device(config)# radius-server host 192.0.2.1 test username test1 idle-time 1	Enables RADIUS automated testing.
Step 4	aaa group server radius <i>group-name</i> Example: Device(config)# aaa group server radius rad-sg	Enters server group configuration mode.
Step 5	load-balance method least-outstanding [batch-size number] [ignore-preferred-server] Example:	Enables the least-outstanding load balancing for a named server group.

	Command or Action	Purpose
	Device(config-sg)# load-balance method least-outstanding batch-size 30	
Step 6	end Example: Device(config-sg)# end	Exits server group configuration mode and enters privileged EXEC mode.

Enabling Load Balancing for a Global RADIUS Server Group

The global RADIUS server group is referred to as “radius” in the authentication, authorization, and accounting (AAA) method lists.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **radius-server host** *{hostname | ip-address}* [**test username** *name*] [**auth-port** *number*] [**ignore-auth-port**] [**acct-port** *number*] [**ignore-acct-port**] [**idle-time** *seconds*]
4. **radius-server load-balance method** **least-outstanding** [**batch-size** *number*] [**ignore-preferred-server**]
5. **load-balance method** **least-outstanding** [**batch-size** *number*] [**ignore-preferred-server**]
6. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	radius-server host <i>{hostname ip-address}</i> [test username <i>name</i>] [auth-port <i>number</i>] [ignore-auth-port] [acct-port <i>number</i>] [ignore-acct-port] [idle-time <i>seconds</i>] Example: Device(config)# radius-server host 192.0.2.1 test username test1 idle-time 1	Enables RADIUS automated testing.
Step 4	radius-server load-balance method least-outstanding [batch-size <i>number</i>] [ignore-preferred-server] Example:	Enables the least-outstanding load balancing for the global RADIUS server group and enters server group configuration mode. <ul style="list-style-type: none"> • The default batch size is 25. The batch size range is from 1 to 2147483647.

	Command or Action	Purpose
	Device(config)# radius-server load-balance method least-outstanding	
Step 5	load-balance method least-outstanding [batch-size <i>number</i>] [ignore-preferred-server] Example: Device(config-sg)# load-balance method least-outstanding batch-size 5	Enables least-outstanding load balancing for a global named server group.
Step 6	end Example: Device(config-sg)# end	Exits server group configuration mode and enters privileged EXEC mode.

Troubleshooting RADIUS Server Load Balancing

After configuring the RADIUS Server Load Balancing feature, you can monitor the idle timer, dead timer, and load balancing server selection or verify the server status by using a manual test command.

SUMMARY STEPS

1. Use the **debug aaa test** command to determine when an idle timer or dead timer has expired, when test packets are sent, the status of the server, or to verify the server state.
2. Use the **debug aaa sg-server selection** command to determine the server that is selected for load balancing.
3. Use the **test aaa group** command to manually verify the RADIUS load-balanced server status.

DETAILED STEPS

Step 1 Use the **debug aaa test** command to determine when an idle timer or dead timer has expired, when test packets are sent, the status of the server, or to verify the server state.

The idle timer is used to check the server status and is updated with or without any incoming requests. Monitoring the idle timer helps to determine if there are nonresponsive servers and to keep the RADIUS server status updated to efficiently utilize available resources. For instance, an updated idle timer would help ensure that incoming requests are sent to servers that are alive.

The dead timer is used either to determine that a server is dead or to update a dead server's status appropriately.

Monitoring server selection helps to determine how often the server selection changes. Server selection is effective in analyzing if there are any bottlenecks, a large number of queued requests, or if only specific servers are processing incoming requests.

The following sample output from the **debug aaa test** command shows when the idle timer expired:

Example:

```
Device# debug aaa test
```

```
Jul 16 00:07:01: AAA/SG/TEST: Server (192.0.2.245:1700,1701) quarantined.
Jul 16 00:07:01: AAA/SG/TEST: Sending test request(s) to server (192.0.2.245:1700,1701)
Jul 16 00:07:01: AAA/SG/TEST: Sending 1 Access-Requests, 1 Accounting-Requests in current batch.
Jul 16 00:07:01: AAA/SG/TEST(Req#: 1): Sending test AAA Access-Request.
Jul 16 00:07:01: AAA/SG/TEST(Req#: 1): Sending test AAA Accounting-Request.
```

```

Jul 16 00:07:01: AAA/SG/TEST: Obtained Test response from server (192.0.2.245:1700,1701)
Jul 16 00:07:01: AAA/SG/TEST: Obtained Test response from server (192.0.2.245:1700,1701)
Jul 16 00:07:01: AAA/SG/TEST: Necessary responses received from server (192.0.2.245:1700,1701)
Jul 16 00:07:01: AAA/SG/TEST: Server (192.0.2.245:1700,1701) marked ALIVE. Idle timer set for 60
sec(s).
Jul 16 00:07:01: AAA/SG/TEST: Server (192.0.2.245:1700,1701) removed from quarantine.

```

Step 2 Use the **debug aaa sg-server selection** command to determine the server that is selected for load balancing.

The following sample output from the **debug aaa sg-server selection** command shows five access requests being sent to a server group with a batch size of three:

Example:

```

Device# debug aaa sg-server selection

Jul 16 03:15:05: AAA/SG/SERVER_SELECT: Obtaining least loaded server.
Jul 16 03:15:05: AAA/SG/SERVER_SELECT: [3] transactions remaining in batch. Reusing server.
Jul 16 03:15:05: AAA/SG/SERVER_SELECT: Obtaining least loaded server.
Jul 16 03:15:05: AAA/SG/SERVER_SELECT: [2] transactions remaining in batch. Reusing server.
Jul 16 03:15:05: AAA/SG/SERVER_SELECT: Obtaining least loaded server.
Jul 16 03:15:05: AAA/SG/SERVER_SELECT: [1] transactions remaining in batch. Reusing server.
Jul 16 03:15:05: AAA/SG/SERVER_SELECT: Obtaining least loaded server.
Jul 16 03:15:05: AAA/SG/SERVER_SELECT: No more transactions in batch. Obtaining a new server.
Jul 16 03:15:05: AAA/SG/SERVER_SELECT: Obtaining a new least loaded server.
Jul 16 03:15:05: AAA/SG/SERVER_SELECT: Server[0] load: 3
Jul 16 03:15:05: AAA/SG/SERVER_SELECT: Server[1] load: 0
Jul 16 03:15:05: AAA/SG/SERVER_SELECT: Server[2] load: 0
Jul 16 03:15:05: AAA/SG/SERVER_SELECT: Selected Server[1] with load 0
Jul 16 03:15:05: AAA/SG/SERVER_SELECT: [3] transactions remaining in batch.
Jul 16 03:15:05: AAA/SG/SERVER_SELECT: Obtaining least loaded server.
Jul 16 03:15:05: AAA/SG/SERVER_SELECT: [2] transactions remaining in batch. Reusing server.

```

Step 3 Use the **test aaa group** command to manually verify the RADIUS load-balanced server status.

The following sample output shows the response from a load-balanced RADIUS server that is alive when the username “test” does not match a user profile. The server is verified alive when it issues an Access-Reject response to an authentication, authorization, and accounting (AAA) packet generated using the **test aaa group** command.

Example:

```

Device# test aaa group SG1 test lab new-code

00:06:07: RADIUS/ENCODE(00000000):Orig. component type = INVALID
00:06:07: RADIUS/ENCODE(00000000): dropping service type, "radius-server attribute 6 on-for-login-auth"
is off
00:06:07: RADIUS(00000000): Config NAS IP: 192.0.2.4
00:06:07: RADIUS(00000000): sending
00:06:07: RADIUS/ENCODE: Best Local IP-Address 192.0.2.141 for Radius-Server 192.0.2.176
00:06:07: RADIUS(00000000): Send Access-Request to 192.0.2.176:1645 id 1645/1, len 50
00:06:07: RADIUS: authenticator CA DB F4 9B 7B 66 C8 A9 - D1 99 4E 8E A4 46 99 B4
00:06:07: RADIUS: User-Password [2] 18 *
00:06:07: RADIUS: User-Name [1] 6 "test"
00:06:07: RADIUS: NAS-IP-Address [4] 6 192.0.2.141
00:06:07: RADIUS: Received from id 1645/1 192.0.2.176:1645, Access-Reject, len 44
00:06:07: RADIUS: authenticator 2F 69 84 3E F0 4E F1 62 - AB B8 75 5B 38 82 49 C3
00:06:07: RADIUS: Reply-Message [18] 24
00:06:07: RADIUS: 41 75 74 68 65 6E 74 69 63 61 74 69 6F 6E 20 66 [Authentication f]
00:06:07: RADIUS: 61 69 6C 75 72 65 [failure]
00:06:07: RADIUS(00000000): Received from id 1645/1
00:06:07: RADIUS/DECODE: Reply-Message fragments, 22, total 22 bytes

```

Configuration Examples for RADIUS Server Load Balancing

Example: Enabling Load Balancing for a Named RADIUS Server Group

The following examples show load balancing enabled for a named RADIUS server group. These examples are shown in three parts: the current configuration of the RADIUS command output, debug output, and authentication, authorization, and accounting (AAA) server status information.

The following sample output shows the relevant RADIUS configuration:

```
Device# show running-config
.
.
.
aaa group server radius server-group1
  server 192.0.2.238 auth-port 2095 acct-port 2096
  server 192.0.2.238 auth-port 2015 acct-port 2016
  load-balance method least-outstanding batch-size 5
!
aaa authentication ppp default group server-group1
aaa accounting network default start-stop group server-group1
.
.
.
```

The lines in the current configuration of the preceding RADIUS command output are defined as follows:

- The **aaa group server radius** command shows the configuration of a server group with two member servers.
- The **load-balance** command enables load balancing for global RADIUS server groups with the batch size specified.
- The **aaa authentication ppp** command authenticates all PPP users using RADIUS.
- The **aaa accounting** command enables sending of all accounting requests to the AAA server when the client is authenticated and then disconnected using the **start-stop** keyword.

The show debug sample output below shows the selection of the preferred server and the processing of requests for the preceding configuration:

```
Device# show debug
*Feb 28 13:51:16.019:AAA/SG/SERVER_SELECT(0000002C):No preferred server available.
*Feb 28 13:51:16.019:AAA/SG/SERVER_SELECT:Obtaining least loaded server.
*Feb 28 13:51:16.019:AAA/SG/SERVER_SELECT:No more transactions in batch. Obtaining a new server.
*Feb 28 13:51:16.019:AAA/SG/SERVER_SELECT:Obtaining a new least loaded server.
*Feb 28 13:51:16.019:AAA/SG/SERVER_SELECT:Server[0] load:0
*Feb 28 13:51:16.019:AAA/SG/SERVER_SELECT:Server[1] load:0
*Feb 28 13:51:16.019:AAA/SG/SERVER_SELECT:Selected Server[0] with load 0
*Feb 28 13:51:16.019:AAA/SG/SERVER_SELECT:[5] transactions remaining in batch.
*Feb 28 13:51:16.019:AAA/SG/SERVER_SELECT(0000002C):Server (192.0.2.238:2095,2096) now being used as preferred server
*Feb 28 13:51:16.019:AAA/SG/SERVER_SELECT(0000002D):No preferred server available.
*Feb 28 13:51:16.019:AAA/SG/SERVER_SELECT:Obtaining least loaded server.
*Feb 28 13:51:16.019:AAA/SG/SERVER_SELECT:[4] transactions remaining in batch. Reusing server.
*Feb 28 13:51:16.019:AAA/SG/SERVER_SELECT(0000002D):Server (192.0.2.238:2095,2096) now being
```

```

used as preferred server
*Feb 28 13:51:16.019:AAA/SG/SERVER_SELECT(0000002E):No preferred server available.
*Feb 28 13:51:16.019:AAA/SG/SERVER_SELECT:Obtaining least loaded server.
*Feb 28 13:51:16.019:AAA/SG/SERVER_SELECT:[3] transactions remaining in batch. Reusing
server.
*Feb 28 13:51:16.019:AAA/SG/SERVER_SELECT(0000002E):Server (192.0.2.238:2095,2096) now being
used as preferred server
*Feb 28 13:51:16.019:AAA/SG/SERVER_SELECT(0000002F):No preferred server available.
*Feb 28 13:51:16.019:AAA/SG/SERVER_SELECT:Obtaining least loaded server.
*Feb 28 13:51:16.019:AAA/SG/SERVER_SELECT:[2] transactions remaining in batch. Reusing
server.
*Feb 28 13:51:16.019:AAA/SG/SERVER_SELECT(0000002F):Server (192.0.2.238:2095,2096) now being
used as preferred server
*Feb 28 13:51:16.019:AAA/SG/SERVER_SELECT(00000030):No preferred server available.
*Feb 28 13:51:16.019:AAA/SG/SERVER_SELECT:Obtaining least loaded server.
*Feb 28 13:51:16.019:AAA/SG/SERVER_SELECT:[1] transactions remaining in batch. Reusing
server.
*Feb 28 13:51:16.019:AAA/SG/SERVER_SELECT(00000030):Server (192.0.2.238:2095,2096) now being
used as preferred server
*Feb 28 13:51:16.023:AAA/SG/SERVER_SELECT(00000031):No preferred server available.
*Feb 28 13:51:16.023:AAA/SG/SERVER_SELECT:Obtaining least loaded server.
*Feb 28 13:51:16.023:AAA/SG/SERVER_SELECT:No more transactions in batch. Obtaining a new
server.
*Feb 28 13:51:16.023:AAA/SG/SERVER_SELECT:Obtaining a new least loaded server.
*Feb 28 13:51:16.023:AAA/SG/SERVER_SELECT:Server[1] load:0
*Feb 28 13:51:16.023:AAA/SG/SERVER_SELECT:Server[0] load:5
*Feb 28 13:51:16.023:AAA/SG/SERVER_SELECT:Selected Server[1] with load 0
*Feb 28 13:51:16.023:AAA/SG/SERVER_SELECT:[5] transactions remaining in batch.
*Feb 28 13:51:16.023:AAA/SG/SERVER_SELECT(00000031):Server (192.0.2.238:2015,2016) now being
used as preferred server
*Feb 28 13:51:16.023:AAA/SG/SERVER_SELECT(00000032):No preferred server available.
*Feb 28 13:51:16.023:AAA/SG/SERVER_SELECT:Obtaining least loaded server.
*Feb 28 13:51:16.023:AAA/SG/SERVER_SELECT:[4] transactions remaining in batch. Reusing
server.
.
.
.

```

The following sample output from the **show aaa servers** command shows the AAA server status for the named RADIUS server group configuration:

The sample output shows the status of two RADIUS servers. Both servers are alive, and no requests have been processed since the counters were cleared 0 minutes ago.

```

Device# show aaa servers

RADIUS:id 8, priority 1, host 192.0.2.238, auth-port 2095, acct-port 2096
  State:current UP, duration 3781s, previous duration 0s
  Dead:total time 0s, count 0
  Quarantined:No
  Authen:request 0, timeouts 0
    Response:unexpected 0, server error 0, incorrect 0, time 0ms
    Transaction:success 0, failure 0
  Author:request 0, timeouts 0
    Response:unexpected 0, server error 0, incorrect 0, time 0ms
    Transaction:success 0, failure 0
  Account:request 0, timeouts 0
    Response:unexpected 0, server error 0, incorrect 0, time 0ms
    Transaction:success 0, failure 0
  Elapsed time since counters last cleared:0m
RADIUS:id 9, priority 2, host 192.0.2.238, auth-port 2015, acct-port 2016
  State:current UP, duration 3781s, previous duration 0s
  Dead:total time 0s, count 0
  Quarantined:No

```

```

Authen:request 0, timeouts 0
      Response:unexpected 0, server error 0, incorrect 0, time 0ms
      Transaction:success 0, failure 0
Author:request 0, timeouts 0
      Response:unexpected 0, server error 0, incorrect 0, time 0ms
      Transaction:success 0, failure 0
Account:request 0, timeouts 0
      Response:unexpected 0, server error 0, incorrect 0, time 0ms
      Transaction:success 0, failure 0
Elapsed time since counters last cleared:0m

```

Example: Enabling Load Balancing for a Global RADIUS Server Group

The following examples show how to enable load balancing for global RADIUS server groups. These examples are shown in three parts: the current configuration of the RADIUS command output, debug output, and authentication, authorization, and accounting (AAA) server status information. You can use delimiting characters to display relevant parts of the configuration.

The following example shows the relevant RADIUS configuration:

```

Device# show running-config | include radius

aaa authentication ppp default group radius
aaa accounting network default start-stop group radius
radius-server host 192.0.2.238 auth-port 2095 acct-port 2096 key cisco
radius-server host 192.0.2.238 auth-port 2015 acct-port 2016 key cisco
radius-server load-balance method least-outstanding batch-size 5

```

Lines in the current configuration of the preceding RADIUS command output are defined as follows:

- The **aaa authentication ppp** command authenticates all PPP users using RADIUS.
- The **aaa accounting** command enables the sending of all accounting requests to an AAA server when the client is authenticated and then disconnected through use of the **start-stop** keyword.
- The **radius-server host** command defines the IP address of the RADIUS server host with the authorization and accounting ports specified and the authentication and encryption keys identified.
- The **radius-server load-balance** command enables load balancing for global RADIUS server groups with the batch size specified.

The **show debug** sample output below shows the selection of the preferred server and the processing of requests for the configuration:

```

Device# show debug

General OS:
  AAA server group server selection debugging is on
#
<sending 10 pppoe requests>
Device#
*Feb 28 13:40:32.199:AAA/SG/SERVER_SELECT(00000014):No preferred server available.
*Feb 28 13:40:32.199:AAA/SG/SERVER_SELECT:Obtaining least loaded server.
*Feb 28 13:40:32.199:AAA/SG/SERVER_SELECT:No more transactions in batch. Obtaining a new
server.
*Feb 28 13:40:32.199:AAA/SG/SERVER_SELECT:Obtaining a new least loaded server.
*Feb 28 13:40:32.199:AAA/SG/SERVER_SELECT:Server[0] load:0
*Feb 28 13:40:32.199:AAA/SG/SERVER_SELECT:Server[1] load:0
*Feb 28 13:40:32.199:AAA/SG/SERVER_SELECT:Selected Server[0] with load 0
*Feb 28 13:40:32.199:AAA/SG/SERVER_SELECT:[5] transactions remaining in batch.
*Feb 28 13:40:32.199:AAA/SG/SERVER_SELECT(00000014):Server (192.0.2.238:2095,2096) now being

```



```

used as preferred server
*Feb 28 13:40:32.199:AAA/SG/SERVER_SELECT(00000015):No preferred server available.
*Feb 28 13:40:32.199:AAA/SG/SERVER_SELECT:Obtaining least loaded server.
*Feb 28 13:40:32.199:AAA/SG/SERVER_SELECT:[4] transactions remaining in batch. Reusing
server.
*Feb 28 13:40:32.199:AAA/SG/SERVER_SELECT(00000015):Server (192.0.2.238:2095,2096) now being
used as preferred server
*Feb 28 13:40:32.199:AAA/SG/SERVER_SELECT(00000016):No preferred server available.
*Feb 28 13:40:32.199:AAA/SG/SERVER_SELECT:Obtaining least loaded server.
*Feb 28 13:40:32.199:AAA/SG/SERVER_SELECT:[3] transactions remaining in batch. Reusing
server.
*Feb 28 13:40:32.199:AAA/SG/SERVER_SELECT(00000016):Server (192.0.2.238:2095,2096) now being
used as preferred server
*Feb 28 13:40:32.199:AAA/SG/SERVER_SELECT(00000017):No preferred server available.
*Feb 28 13:40:32.199:AAA/SG/SERVER_SELECT:Obtaining least loaded server.
*Feb 28 13:40:32.199:AAA/SG/SERVER_SELECT:[2] transactions remaining in batch. Reusing
server.
*Feb 28 13:40:32.199:AAA/SG/SERVER_SELECT(00000017):Server (192.0.2.238:2095,2096) now being
used as preferred server
*Feb 28 13:40:32.199:AAA/SG/SERVER_SELECT(00000018):No preferred server available.
*Feb 28 13:40:32.199:AAA/SG/SERVER_SELECT:Obtaining least loaded server.
*Feb 28 13:40:32.199:AAA/SG/SERVER_SELECT:[1] transactions remaining in batch. Reusing
server.
*Feb 28 13:40:32.199:AAA/SG/SERVER_SELECT(00000018):Server (192.0.2.238:2095,2096) now being
used as preferred server
*Feb 28 13:40:32.199:AAA/SG/SERVER_SELECT(00000019):No preferred server available.
*Feb 28 13:40:32.199:AAA/SG/SERVER_SELECT:Obtaining least loaded server.
*Feb 28 13:40:32.199:AAA/SG/SERVER_SELECT:No more transactions in batch. Obtaining a new
server.
*Feb 28 13:40:32.199:AAA/SG/SERVER_SELECT:Obtaining a new least loaded server.
*Feb 28 13:40:32.199:AAA/SG/SERVER_SELECT:Server[1] load:0
*Feb 28 13:40:32.199:AAA/SG/SERVER_SELECT:Server[0] load:5
*Feb 28 13:40:32.199:AAA/SG/SERVER_SELECT:Selected Server[1] with load 0
*Feb 28 13:40:32.199:AAA/SG/SERVER_SELECT:[5] transactions remaining in batch.
*Feb 28 13:40:32.199:AAA/SG/SERVER_SELECT(00000019):Server (192.0.2.238:2015,2016) now being
used as preferred server.

```

The following sample output from the **show aaa servers** command shows the AAA server status for the global RADIUS server group configuration:

The sample output shows the status of two RADIUS servers. Both servers are up and successfully processed in the last 2 minutes:

- Five out of six authentication requests
- Five out of five accounting requests

Device# **show aaa servers**

```

RADIUS:id 4, priority 1, host 192.0.2.238, auth-port 2095, acct-port 2096
  State:current UP, duration 3175s, previous duration 0s
  Dead:total time 0s, count 0
  Quarantined:No
  Authen:request 6, timeouts 1
    Response:unexpected 1, server error 0, incorrect 0, time 1841ms
    Transaction:success 5, failure 0
  Author:request 0, timeouts 0
    Response:unexpected 0, server error 0, incorrect 0, time 0ms
    Transaction:success 0, failure 0
  Account:request 5, timeouts 0
    Response:unexpected 0, server error 0, incorrect 0, time 3303ms
    Transaction:success 5, failure 0
  Elapsed time since counters last cleared:2m
RADIUS:id 5, priority 2, host 192.0.2.238, auth-port 2015, acct-port 2016

```

```

State:current UP, duration 3175s, previous duration 0s
Dead:total time 0s, count 0
Quarantined:No
Authen:request 6, timeouts 1
      Response:unexpected 1, server error 0, incorrect 0, time 1955ms
      Transaction:success 5, failure 0
Author:request 0, timeouts 0
      Response:unexpected 0, server error 0, incorrect 0, time 0ms
      Transaction:success 0, failure 0
Account:request 5, timeouts 0
      Response:unexpected 0, server error 0, incorrect 0, time 3247ms
      Transaction:success 5, failure 0
Elapsed time since counters last cleared:2m

```

Example: Monitoring Idle Timer

The following example shows idle timer and related server state for load balancing enabled for a named RADIUS server group. The current configuration of the RADIUS command output and debug command output are also displayed.

The following sample output shows the relevant RADIUS configuration:

```

Device# show running-config | include radius

aaa group server radius server-group1
radius-server host 192.0.2.238 auth-port 2095 acct-port 2096 test username junk1 idle-time
 1 key cisco
radius-server host 192.0.2.238 auth-port 2015 acct-port 2016 test username junk1 idle-time
 1 key cisco
radius-server load-balance method least-outstanding batch-size 5

```

The lines in the current configuration of the preceding RADIUS command output are defined as follows:

- The **aaa group server radius** command shows the configuration of a server group.
- The **radius-server host** command defines the IP address of the RADIUS server host with authorization and accounting ports specified and the authentication and encryption key identified.
- The **radius-server load-balance** command enables load balancing for the RADIUS server with the batch size specified.

The **show debug** sample output below shows test requests being sent to servers. The response to the test request sent to the server is received, the server is removed from quarantine as appropriate, the server is marked alive, and then the idle timer is reset.

```

Device# show debug

*Feb 28 13:52:20.835:AAA/SG/TEST:Server (192.0.2.238:2015,2016) quarantined.
*Feb 28 13:52:20.835:AAA/SG/TEST:Sending test request(s) to server (192.0.2.238:2015,2016)
*Feb 28 13:52:20.835:AAA/SG/TEST:Sending 1 Access-Requests, 1 Accounting-Requests in current
  batch.
*Feb 28 13:52:20.835:AAA/SG/TEST(Req#:1):Sending test AAA Access-Request.
*Feb 28 13:52:20.835:AAA/SG/TEST(Req#:1):Sending test AAA Accounting-Request.
*Feb 28 13:52:21.087:AAA/SG/TEST:Obtained Test response from server (192.0.2.238:2015,2016)
*Feb 28 13:52:22.651:AAA/SG/TEST:Obtained Test response from server (192.0.2.238:2015,2016)
*Feb 28 13:52:22.651:AAA/SG/TEST:Necessary responses received from server
(192.0.2.238:2015,2016)
*Feb 28 13:52:22.651:AAA/SG/TEST:Server (192.0.2.238:2015,2016) marked ALIVE. Idle timer
set for 60 secs(s).
*Feb 28 13:52:22.651:AAA/SG/TEST:Server (192.0.2.238:2015,2016) removed from quarantine.
.

```

Example: Configuring the Preferred Server with the Same Authentication and Authorization Server

The following example shows an authentication server group and an authorization server group that use the same servers 209.165.200.225 and 209.165.200.226. Both server groups have the preferred server flag enabled.

```
aaa group server radius authentication-group
  server 209.165.200.225 key radkey1
  server 209.165.200.226 key radkey2
aaa group server radius accounting-group
  server 209.165.200.225 key radkey1
  server 209.165.200.226 key radkey2
```

When a preferred server is selected for a session, all transactions for that session will continue to use the original preferred server. The servers 209.165.200.225 and 209.165.200.226 are load balanced based on sessions rather than transactions.

Example: Configuring the Preferred Server with Different Authentication and Authorization Servers

The following example shows an authentication server group that uses servers 209.165.200.225 and 209.165.200.226 and an authorization server group that uses servers 209.165.201.1 and 209.165.201.2. Both server groups have the preferred server flag enabled.

```
aaa group server radius authentication-group
  server 209.165.200.225 key radkey1
  server 209.165.200.226 key radkey2
aaa group server radius accounting-group
  server 209.165.201.1 key radkey3
  server 209.165.201.2 key radkey4
```

The authentication server group and the accounting server group do not share any common servers. A preferred server is never found for accounting transactions; therefore, authentication and accounting servers are load-balanced based on transactions. Start and stop records are sent to the same server for a session.

Example: Configuring the Preferred Server with Overlapping Authentication and Authorization Servers

The following example shows an authentication server group that uses servers 209.165.200.225, 209.165.200.226, and 209.165.201.1 and an accounting server group that uses servers 209.165.201.1 and 209.165.201.2. Both server groups have the preferred server flag enabled.

```
aaa group server radius authentication-group
  server 209.165.200.225 key radkey1
  server 209.165.200.226 key radkey2
  server 209.165.201.1 key radkey3
aaa group server radius accounting-group
  server 209.165.201.1 key radkey3
  server 209.165.201.2 key radkey4
```

If all servers have equal transaction processing capability, one-third of all authentication transactions are directed toward the server 209.165.201.1. Therefore, one-third of all accounting transactions are also directed

toward the server 209.165.201.1. The remaining two-third of accounting transactions are load balanced equally between servers 209.165.201.1 and 209.165.201.2. The server 209.165.201.1 receives fewer authentication transactions because the server 209.165.201.1 has outstanding accounting transactions.

Example: Configuring the Preferred Server with Authentication Servers As a Subset of Authorization Servers

The following example shows an authentication server group that uses servers 209.165.200.225 and 209.165.200.226 and an authorization server group that uses servers 209.165.200.225, 209.165.200.226, and 209.165.201.1. Both server groups have the preferred server flag enabled.

```
aaa group server radius authentication-group
  server 209.165.200.225 key radkey1
  server 209.165.200.226 key radkey2
aaa group server radius accounting-group
  server 209.165.200.225 key radkey1
  server 209.165.200.226 key radkey2
  server 209.165.201.1 key radkey3
```

One-half of all authentication transactions are sent to the server 209.165.200.225 and the other half to the server 209.165.200.226. Servers 209.165.200.225 and 209.165.200.226 are preferred servers for authentication and accounting transaction. Therefore, there is an equal distribution of authentication and accounting transactions across servers 209.165.200.225 and 209.165.200.226. The server 209.165.201.1 is relatively unused.

Example: Configuring the Preferred Server with Authentication Servers As a Superset of Authorization Servers

The following example shows an authentication server group that uses servers 209.165.200.225, 209.165.200.226, and 209.165.201.1 and an authorization server group that uses servers 209.165.200.225 and 209.165.200.226. Both server groups have the preferred server flag enabled.

```
aaa group server radius authentication-group
  server 209.165.200.225 key radkey1
  server 209.165.200.226 key radkey2
  server 209.165.201.1 key radkey3
aaa group server radius accounting-group
  server 209.165.200.225 key radkey1
  server 209.165.200.226 key radkey2
```

Initially, one-third of authentication transactions are assigned to each server in the authorization server group. As accounting transactions are generated for more sessions, accounting transactions are sent to servers 209.165.200.225 and 209.165.200.226 because the preferred server flag is on. As servers 209.165.200.225 and 209.165.200.226 begin to process more transactions, authentication transactions will start to be sent to server 209.165.201.1. Transaction requests authenticated by server 209.165.201.1 do not have any preferred server setting and are split between servers 209.165.200.225 and 209.165.200.226, which negates the use of the preferred server flag. This configuration should be used cautiously.

Additional References for RADIUS Server Load Balancing

Related Documents

Related Topic	Document Title
Security commands	<ul style="list-style-type: none"> • Security Command Reference: Commands A to C • Security Command Reference: Commands D to L • Security Command Reference: Commands M to R • Security Command Reference: Commands S to Z

Technical Assistance

Description	Link
<p>The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies.</p> <p>To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds.</p> <p>Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.</p>	http://www.cisco.com/techsupport

Feature Information for RADIUS Server Load Balancing

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Table 114: Feature Information for RADIUS Server Load Balancing

Feature Name	Releases	Feature Information
RADIUS Server Load Balancing	Cisco IOS 15.2(1)E	<p>The RADIUS Server Load Balancing feature distributes authentication, authorization, and accounting (AAA) authentication and accounting transactions across servers in a server group. These servers can share the AAA transaction load and thereby respond faster to incoming requests.</p> <p>The following commands were introduced or modified: debug aaa sg-server selection, debug aaa test, load-balance (server-group), radius-server host, radius-server load-balance, and test aaa group.</p>



CHAPTER 53

RADIUS Change of Authorization Support

The RADIUS Change of Authorization (CoA) provides a mechanism to change the attributes of an authentication, authorization, and accounting (AAA) session after it is authenticated

Identity-Based Networking Services supports RADIUS change of authorization (CoA) commands for session query, reauthentication, and termination, port bounce and port shutdown, and service template activation and deactivation.

- [Information About RADIUS Change-of-Authorization, on page 1117](#)
- [How to Configure RADIUS Change-of-Authorization, on page 1128](#)
- [Additional References for RADIUS Change-of-Authorization, on page 1131](#)
- [Feature Information for RADIUS Change-of-Authorization Support, on page 1131](#)

Information About RADIUS Change-of-Authorization

RADIUS Change of Authorization

The RADIUS Change of Authorization (CoA) provides a mechanism to change the attributes of an authentication, authorization, and accounting (AAA) session after it is authenticated. When a policy changes for a user or user group in AAA, administrators can send RADIUS CoA packets from the AAA server such as a Cisco Secure Access Control Server (ACS) to reinitialize authentication and apply the new policy. This section provides an overview of the RADIUS interface including available primitives and how they are used during a CoA.

- Change-of-Authorization Requests
- CoA Request Response Code
- CoA Request Commands
- Session Reauthentication
- Stacking Guidelines for Session Termination

A standard RADIUS interface is typically used in a pulled model where the request originates from a network attached device and the response come from the queried servers. Catalyst support the RADIUS CoA extensions defined in RFC 5176 that are typically used in a pushed model and allow for the dynamic reconfiguring of sessions from external AAA or policy servers.

The supports these per-session CoA requests:

- Session reauthentication
- Session termination
- Session termination with port shutdown
- Session termination with port bounce

This feature is integrated with Cisco Secure Access Control Server (ACS) 5.1.

The RADIUS interface is enabled by default on Catalyst . However, some basic configuration is required for the following attributes:

- Security and Password—refer to the “Preventing Unauthorized Access to Your Switch” section in this guide.
- Accounting—refer to the “Starting RADIUS Accounting” section in the Configuring Switch-Based Authentication chapter in this guide.

Cisco IOS software supports the RADIUS CoA extensions defined in RFC 5176 that are typically used in a push model to allow the dynamic reconfiguring of sessions from external AAA or policy servers. Per-session CoA requests are supported for session identification, session termination, host reauthentication, port shutdown, and port bounce. This model comprises one request (CoA-Request) and two possible response codes:

- CoA acknowledgement (ACK) [CoA-ACK]
- CoA nonacknowledgement (NAK) [CoA-NAK]

The request is initiated from a CoA client (typically a AAA or policy server) and directed to the device that acts as a listener.

The table below shows the RADIUS CoA commands and vendor-specific attributes (VSAs) supported by Identity-Based Networking Services. All CoA commands must include the session identifier between the device and the CoA client.

Table 115: RADIUS CoA Commands Supported by Identity-Based Networking Services

CoA Command	Cisco VSA
Activate service	Cisco:Avpair=“subscriber:command=activate-service” Cisco:Avpair=“subscriber:service-name=<service-name>” Cisco:Avpair=“subscriber:precedence=<precedence-number>” Cisco:Avpair=“subscriber:activation-mode=replace-all”
Deactivate service	Cisco:Avpair=“subscriber:command=deactivate-service” Cisco:Avpair=“subscriber:service-name=<service-name>”
Bounce host port	Cisco:Avpair=“subscriber:command=bounce-host-port”
Disable host port	Cisco:Avpair=“subscriber:command=disable-host-port”
Session query	Cisco:Avpair=“subscriber:command=session-query”

CoA Command	Cisco VSA
Session reauthenticate	Cisco:Avpair="subscriber:command=reauthenticate" Cisco:Avpair="subscriber:reauthenticate-type=last" or Cisco:Avpair="subscriber:reauthenticate-type=rerun"
Session terminate	This is a standard disconnect request and does not require a VSA.
Interface template	Cisco:AVpair="interface-template-name=<interfacetemplate>"

Change-of-Authorization Requests

Change of Authorization (CoA) requests, as described in RFC 5176, are used in a push model to allow for session identification, host reauthentication, and session termination. The model is comprised of one request (CoA-Request) and two possible response codes:

- CoA acknowledgment (ACK) [CoA-ACK]
- CoA non-acknowledgment (NAK) [CoA-NAK]

The request is initiated from a CoA client (typically a RADIUS or policy server) and directed to the switch that acts as a listener.

RFC 5176 Compliance

The Disconnect Request message, which is also referred to as Packet of Disconnect (POD), is supported by the switch for session termination.

This table shows the IETF attributes are supported for this feature.

Table 116: Supported IETF Attributes

Attribute Number	Attribute Name
24	State
31	Calling-Station-ID
44	Acct-Session-ID
80	Message-Authenticator
101	Error-Cause

This table shows the possible values for the Error-Cause attribute.

Table 117: Error-Cause Values

Value	Explanation
201	Residual Session Context Removed

Value	Explanation
202	Invalid EAP Packet (Ignored)
401	Unsupported Attribute
402	Missing Attribute
403	NAS Identification Mismatch
404	Invalid Request
405	Unsupported Service
406	Unsupported Extension
407	Invalid Attribute Value
501	Administratively Prohibited
502	Request Not Routable (Proxy)
503	Session Context Not Found
504	Session Context Not Removable
505	Other Proxy Processing Error
506	Resources Unavailable
507	Request Initiated
508	Multiple Session Selection Unsupported

Preconditions

To use the CoA interface, a session must already exist on the switch. CoA can be used to identify a session and enforce a disconnect request. The update affects only the specified session.

CoA Request Response Code

The CoA Request response code can be used to convey a command to the switch.

The packet format for a CoA Request Response code as defined in RFC 5176 consists of the following fields: Code, Identifier, Length, Authenticator, and Attributes in the Type:Length:Value (TLV) format. The Attributes field is used to carry Cisco vendor-specific attributes (VSAs).

Session Identification

For disconnect and CoA requests targeted at a particular session, the switch locates the session based on one or more of the following attributes:

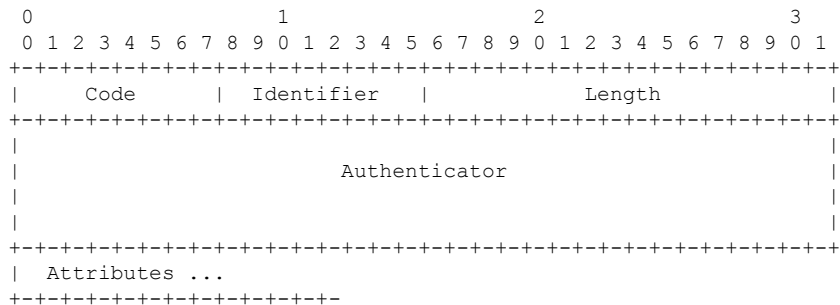
- Acct-Session-Id (IETF attribute #44)

- Audit-Session-Id (Cisco VSA)
- Calling-Station-Id (IETF attribute #31 which contains the host MAC address)
- IPv6 Attributes, which can be one of the following:
 - Framed-IPv6-Prefix (IETF attribute #97) and Framed-Interface-Id (IETF attribute #96), which together create a full IPv6 address per RFC 3162
 - Framed-IPv6-Address
- Plain IP Address (IETF attribute #8)

Unless all session identification attributes included in the CoA message match the session, the switch returns a Disconnect-NAK or CoA-NAK with the “Invalid Attribute Value” error-code attribute.

If more than one session identification attribute is included in the message, all the attributes must match the session or the switch returns a Disconnect- negative acknowledgment (NAK) or CoA-NAK with the error code “Invalid Attribute Value.”

The packet format for a CoA Request code as defined in RFC 5176 consists of the fields: Code, Identifier, Length, Authenticator, and Attributes in Type:Length:Value (TLV) format.



The attributes field is used to carry Cisco vendor-specific attributes (VSAs).

For CoA requests targeted at a particular enforcement policy, the device returns a CoA-NAK with the error code “Invalid Attribute Value” if any of the above session identification attributes are included in the message.

Session Identification

For disconnect and CoA requests targeted at a particular session, the device locates the session based on one or more of the following attributes:

- Acct-Session-Id (IETF attribute #44)
- Audit-Session-Id (Cisco VSA)
- Calling-Station-Id (IETF attribute #31, which contains the host MAC address)
- IPv6 Attributes, which can be one of the following:
 - Framed-IPv6-Prefix (IETF attribute #97) and Framed-Interface-Id (IETF attribute #96), which together create a full IPv6 address per RFC 3162
 - Framed-IPv6-Address

- Plain IP Address (IETF attribute #8)

If more than one session identification attribute is included in the message, all of the attributes must match the session or the device returns a Disconnect-NAK or CoA-NAK with the error code “Invalid Attribute Value.”

For CoA requests targeted at a particular enforcement policy, the device returns a CoA-NAK with the error code “Invalid Attribute Value” if any of the above session identification attributes are included in the message.

CoA ACK Response Code

If the authorization state is changed successfully, a positive acknowledgment (ACK) is sent. The attributes returned within CoA ACK will vary based on the CoA Request and are discussed in individual CoA Commands.

CoA NAK Response Code

A negative acknowledgment (NAK) indicates a failure to change the authorization state and can include attributes that indicate the reason for the failure. Use **show** commands to verify a successful CoA.

Session Reauthentication

The AAA server typically generates a session reauthentication request when a host with an unknown identity or posture joins the network and is associated with a restricted access authorization profile (such as a guest VLAN). A reauthentication request allows the host to be placed in the appropriate authorization group when its credentials are known.

To initiate session authentication, the AAA server sends a standard CoA-Request message which contains a Cisco VSA in this form: *Cisco:Avpair=“subscriber:command=reauthenticate”* and one or more session identification attributes.

The current session state determines the switch response to the message. If the session is currently authenticated by IEEE 802.1x, the switch responds by sending an EAPoL (Extensible Authentication Protocol over Lan) -RequestId message to the server.

If the session is currently authenticated by MAC authentication bypass (MAB), the switch sends an access-request to the server, passing the same identity attributes used for the initial successful authentication.

If session authentication is in progress when the switch receives the command, the switch terminates the process, and restarts the authentication sequence, starting with the method configured to be attempted first.

If the session is not yet authorized, or is authorized via guest VLAN, or critical VLAN, or similar policies, the reauthentication message restarts the access control methods, beginning with the method configured to be attempted first. The current authorization of the session is maintained until the reauthentication leads to a different authorization result.

Session Reauthentication in a Switch Stack

When a switch stack receives a session reauthentication message:

- It checkpoints the need for a re-authentication before returning an acknowledgment (ACK).
- It initiates reauthentication for the appropriate session.

- If authentication completes with either success or failure, the signal that triggered the reauthentication is removed from the stack's member switch.
- If the stack's active switch fails before authentication completes, reauthentication is initiated after active switch changeover based on the original command (which is subsequently removed).
- If the active switch fails before sending an ACK, the new active switch treats the re-transmitted command as a new command.

Session Termination

There are three types of CoA requests that can trigger session termination. A CoA Disconnect-Request terminates the session, without disabling the host port. This command causes re-initialization of the authenticator state machine for the specified host, but does not restrict that host access to the network.

To restrict a host's access to the network, use a CoA Request with the Cisco:Avpair="subscriber:command=disable-host-port" VSA. This command is useful when a host is known to be causing problems on the network, and you need to immediately block network access for the host. When you want to restore network access on the port, re-enable it using a non-RADIUS mechanism.

When a device with no supplicant, such as a printer, needs to acquire a new IP address (for example, after a VLAN change), terminate the session on the host port with port-bounce (temporarily disable and then re-enable the port).

CoA Activate Service Command

The CoA activate service command can be used to activate a service template on a session. The AAA server sends the request in a standard CoA-Request message using the following VSAs:

```
Cisco:Avpair="subscriber:command=activate-service"
```

```
Cisco:Avpair="subscriber:service-name=<service-name>"
```

```
Cisco:Avpair="subscriber:precedence=<precedence-number>"
```

```
Cisco:Avpair="subscriber:activation-mode=replace-all"
```

Because this command is session-oriented, it must be accompanied by one or more of the session identification attributes described in the *Session Identification* section below. If the device cannot locate a session, it returns a CoA-NAK message with the "Session Context Not Found" error-code attribute. If the device locates a session, it initiates an activate template operation for the hosting port and a CoA-ACK is returned. If activating the template fails, a CoA-NAK message is returned with the Error-Code attribute set to the appropriate message.

If the device fails before returning a CoA-ACK to the client, the process is repeated on the new active device when the request is re-sent from the client. If the device fails after returning a CoA-ACK message to the client but before the operation is complete, the operation is restarted on the new active device.

Session Identification

For disconnect and CoA requests targeted at a particular session, the device locates the session based on one or more of the following attributes:

- Acct-Session-Id (IETF attribute #44)
- Audit-Session-Id (Cisco VSA)

- Calling-Station-Id (IETF attribute #31, which contains the host MAC address)
- IPv6 Attributes, which can be one of the following:
 - Framed-IPv6-Prefix (IETF attribute #97) and Framed-Interface-Id (IETF attribute #96), which together create a full IPv6 address per RFC 3162
 - Framed-IPv6-Address
- Plain IP Address (IETF attribute #8)

If more than one session identification attribute is included in the message, all of the attributes must match the session or the device returns a Disconnect-NAK or CoA-NAK with the error code “Invalid Attribute Value.”

For CoA requests targeted at a particular enforcement policy, the device returns a CoA-NAK with the error code “Invalid Attribute Value” if any of the above session identification attributes are included in the message.

CoA Deactivate Service Command

The CoA deactivate service command can be used to deactivate a service template on a session. The AAA server sends the request in a standard CoA-Request message using the following VSAs:

```
Cisco:Avpair="subscriber:command=deactivate-service"
```

```
Cisco:Avpair="subscriber:service-name=<service-name>"
```

Because this command is session-oriented, it must be accompanied by one or more of the session identification attributes described in the *Session Identification* section below. If the device cannot locate a session, it returns a CoA-NAK message with the “Session Context Not Found” error-code attribute. If the device locates a session, it initiates a deactivate template operation for the hosting port and a CoA-ACK is returned. If deactivating the template fails, a CoA-NAK message is returned with the Error-Code attribute set to the appropriate message.

If the device fails before returning a CoA-ACK to the client, the process is repeated on the new active device when the request is re-sent from the client. If the device fails after returning a CoA-ACK message to the client but before the operation is complete, the operation is restarted on the new active device.

Session Identification

For disconnect and CoA requests targeted at a particular session, the device locates the session based on one or more of the following attributes:

- Acct-Session-Id (IETF attribute #44)
- Audit-Session-Id (Cisco VSA)
- Calling-Station-Id (IETF attribute #31, which contains the host MAC address)
- IPv6 Attributes, which can be one of the following:
 - Framed-IPv6-Prefix (IETF attribute #97) and Framed-Interface-Id (IETF attribute #96), which together create a full IPv6 address per RFC 3162
 - Framed-IPv6-Address
- Plain IP Address (IETF attribute #8)

If more than one session identification attribute is included in the message, all of the attributes must match the session or the device returns a Disconnect-NAK or CoA-NAK with the error code “Invalid Attribute Value.”

For CoA requests targeted at a particular enforcement policy, the device returns a CoA-NAK with the error code “Invalid Attribute Value” if any of the above session identification attributes are included in the message.

CoA Request: Disable Host Port

The RADIUS server CoA disable port command administratively shuts down the authentication port that is hosting a session, resulting in session termination. This command is useful when a host is known to cause problems on the network and network access needs to be immediately blocked for the host. To restore network access on the port, reenable it using a non-RADIUS mechanism. This command is carried in a standard CoA-Request message that has this new vendor-specific attribute (VSA):

```
Cisco:Avpair="subscriber:command=disable-host-port"
```

Because this command is session-oriented, it must be accompanied by one or more of the session identification attributes described in the “Session Identification” section. If the session cannot be located, the switch returns a CoA-NAK message with the “Session Context Not Found” error-code attribute. If the session is located, the switch disables the hosting port and returns a CoA-ACK message.

If the switch fails before returning a CoA-ACK to the client, the process is repeated on the new active switch when the request is re-sent from the client. If the switch fails after returning a CoA-ACK message to the client but before the operation has completed, the operation is restarted on the new active switch.



Note A Disconnect-Request failure following command re-sending could be the result of either a successful session termination before change-over (if the Disconnect-ACK was not sent) or a session termination by other means (for example, a link failure) that occurred after the original command was issued and before the standby switch became active.

CoA Request: Bounce-Port

A RADIUS server CoA bounce port sent from a RADIUS server can cause a link flap on an authentication port, which triggers DHCP renegotiation from one or more hosts connected to this port. This incident can occur when there is a VLAN change and the endpoint is a device (such as a printer) that does not have a mechanism to detect a change on this authentication port. The CoA bounce port is carried in a standard CoA-Request message that contains the following VSA:

```
Cisco:Avpair="subscriber:command=bounce-host-port"
```

Because this command is session-oriented, it must be accompanied by one or more of the session identification attributes. If the session cannot be located, the switch returns a CoA-NAK message with the “Session Context Not Found” error-code attribute. If the session is located, the switch disables the hosting port for a period of 10 seconds, re-enables it (port-bounce), and returns a CoA-ACK.

If the switch fails before returning a CoA-ACK to the client, the process is repeated on the new active switch when the request is re-sent from the client. If the switch fails after returning a CoA-ACK message to the client but before the operation has completed, the operation is re-started on the new active switch.

CoA Session Query Command

The CoA session query command requests service information about a subscriber session. The AAA server sends the request in a standard CoA-Request message containing the following VSA:

```
Cisco:Avpair="subscriber:command=session-query"
```

Because this command is session-oriented, it must be accompanied by one or more of the session identification attributes described in the *Session Identification* section below. If the device cannot locate a session, it returns a CoA-NAK message with the "Session Context Not Found" error-code attribute. If the device locates a session, it performs a session query operation on the session and returns a CoA-ACK message.

If the device fails before returning a CoA-ACK to the client, the process is repeated on the new active device when the request is re-sent from the client. If the device fails after returning a CoA-ACK message to the client but before the operation is complete, the operation is restarted on the new active device.

Session Identification

For disconnect and CoA requests targeted at a particular session, the device locates the session based on one or more of the following attributes:

- Acct-Session-Id (IETF attribute #44)
- Audit-Session-Id (Cisco VSA)
- Calling-Station-Id (IETF attribute #31, which contains the host MAC address)
- IPv6 Attributes, which can be one of the following:
 - Framed-IPv6-Prefix (IETF attribute #97) and Framed-Interface-Id (IETF attribute #96), which together create a full IPv6 address per RFC 3162
 - Framed-IPv6-Address
- Plain IP Address (IETF attribute #8)

If more than one session identification attribute is included in the message, all of the attributes must match the session or the device returns a Disconnect-NAK or CoA-NAK with the error code "Invalid Attribute Value."

For CoA requests targeted at a particular enforcement policy, the device returns a CoA-NAK with the error code "Invalid Attribute Value" if any of the above session identification attributes are included in the message.

CoA Session Reauthenticate Command

To initiate session authentication, the AAA server sends a standard CoA-Request message containing the following VSAs:

```
Cisco:Avpair="subscriber:command=reauthenticate"
```

```
Cisco:Avpair="subscriber:reauthenticate-type=<last | rerun>"
```

"reauthenticate-type" defines whether the CoA reauthentication request uses the authentication method that last succeeded on the session or whether the authentication process is completely rerun.

The following rules apply:

- "subscriber:command=reauthenticate" must be present to trigger a reauthentication.

- If “subscriber:reauthenticate-type” is not specified, the default behavior is to rerun the last successful authentication method for the session. If the method reauthenticates successfully, all old authorization data is replaced with the new reauthenticated authorization data.
- “subscriber:reauthenticate-type” is valid only when included with “subscriber:command=reauthenticate.” If it is included in another CoA command, the VSA will be silently ignored.

If the device fails before returning a CoA-ACK to the client, the process is repeated on the new active device when the request is resent from the client. If the device fails after returning a CoA-ACK message to the client but before the operation is complete, the operation is restarted on the new active device.

CoA Session Terminate Command

A CoA Disconnect-Request command terminates a session without disabling the host port. This command causes reinitialization of the authenticator state machine for the specified host, but does not restrict the host’s access to the network. If the session cannot be located, the device returns a Disconnect-NAK message with the “Session Context Not Found” error-code attribute. If the session is located, the device terminates the session. After the session has been completely removed, the device returns a Disconnect-ACK.

If the device fails before returning a CoA-ACK to the client, the process is repeated on the new active device when the request is re-sent from the client.

To restrict a host’s access to the network, use a CoA Request with the Cisco:Avpair=“subscriber:command=disable-host-port” VSA. This command is useful when a host is known to cause problems on the network and network access needs to be immediately blocked for the host. When you want to restore network access on the port, reenable it using a non-RADIUS mechanism.

Stacking Guidelines for Session Termination

No special handling is required for CoA Disconnect-Request messages in a switch stack.

Stacking Guidelines for CoA-Request Bounce-Port

Because the **bounce-port** command is targeted at a session, not a port, if the session is not found, the command cannot be executed.

When the Auth Manager command handler on the active switch receives a valid **bounce-port** command, it checkpoints the following information before returning a CoA-ACK message:

- the need for a port-bounce
- the port-id (found in the local session context)

The switch initiates a port-bounce (disables the port for 10 seconds, then re-enables it).

If the port-bounce is successful, the signal that triggered the port-bounce is removed from the standby switch.

If the active switch fails before the port-bounce completes, a port-bounce is initiated after an active switch changeover based on the original command (which is subsequently removed).

If the active switch fails before sending a CoA-ACK message, the new active switch treats the re-sent command as a new command.

Stacking Guidelines for CoA-Request Disable-Port

Because the **disable-port** command is targeted at a session, not a port, if the session is not found, the command cannot be executed.

When the Auth Manager command handler on the active switch receives a valid **disable-port** command, it verifies this information before returning a CoA-ACK message:

- the need for a port-disable
- the port-id (found in the local session context)

The switch attempts to disable the port.

If the port-disable operation is successful, the signal that triggered the port-disable is removed from the standby switch.

If the active switch fails before the port-disable operation completes, the port is disabled after an active switch changeover based on the original command (which is subsequently removed).

If the active switch fails before sending a CoA-ACK message, the new active switch treats the re-sent command as a new command.

How to Configure RADIUS Change-of-Authorization

Configuring CoA on the Device

Follow these steps to configure CoA on a device. This procedure is required.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **aaa new-model**
4. **aaa server radius dynamic-author**
5. **client** {*ip-address* | *name*} [**vrf** *vrfname*] [**server-key** *string*]
6. **server-key** [**0** | **7**] *string*
7. **port** *port-number*
8. **auth-type** {*any* | *all* | *session-key*}
9. **ignore session-key**
10. **ignore server-key**
11. **authentication command bounce-port ignore**
12. **authentication command disable-port ignore**
13. **end**
14. **show running-config**
15. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	aaa new-model Example: Device(config)# aaa new-model	Enables AAA.
Step 4	aaa server radius dynamic-author Example: Device(config)# aaa server radius dynamic-author	Configures the device as an authentication, authorization, and accounting (AAA) server to facilitate interaction with an external policy server.
Step 5	client {ip-address name} [vrf vrfname] [server-key string]	Enters dynamic authorization local server configuration mode and specifies a RADIUS client from which a device will accept CoA and disconnect requests.
Step 6	server-key [0 7] string Example: Device(config-sg-radius)# server-key your_server_key	Configures the RADIUS key to be shared between a device and RADIUS clients.
Step 7	port port-number Example: Device(config-sg-radius)# port 25	Specifies the port on which a device listens for RADIUS requests from configured RADIUS clients.
Step 8	auth-type {any all session-key} Example: Device(config-sg-radius)# auth-type any	Specifies the type of authorization the device uses for RADIUS clients. The client must match all the configured attributes for authorization.
Step 9	ignore session-key	(Optional) Configures the device to ignore the session-key.

	Command or Action	Purpose
		For more information about the ignore command, see the <i>Cisco IOS Intelligent Services Gateway Command Reference</i> on Cisco.com.
Step 10	ignore server-key Example: Device(config-sg-radius)# ignore server-key	(Optional) Configures the device to ignore the server-key. For more information about the ignore command, see the <i>Cisco IOS Intelligent Services Gateway Command Reference</i> on Cisco.com.
Step 11	authentication command bounce-port ignore Example: Device(config-sg-radius)# authentication command bounce-port ignore	(Optional) Configures the device to ignore a CoA request to temporarily disable the port hosting a session. The purpose of temporarily disabling the port is to trigger a DHCP renegotiation from the host when a VLAN change occurs and there is no supplicant on the endpoint to detect the change.
Step 12	authentication command disable-port ignore Example: Device(config-sg-radius)# authentication command disable-port ignore	(Optional) Configures the device to ignore a nonstandard command requesting that the port hosting a session be administratively shut down. Shutting down the port results in termination of the session. Use standard CLI or SNMP commands to re-enable the port.
Step 13	end Example: Device(config-sg-radius)# end	Returns to privileged EXEC mode.
Step 14	show running-config Example: Device# show running-config	Verifies your entries.
Step 15	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Monitoring and Troubleshooting CoA Functionality

The following Cisco IOS commands can be used to monitor and troubleshoot CoA functionality on the switch:

- **debug radius**
- **debug aaa coa**

- `debug aaa pod`
- `debug aaa subsys`
- `debug cmdhd [detail | error | events]`
- `show aaa attributes protocol radius`

Additional References for RADIUS Change-of-Authorization

Related Documents

Related Topic	Document Title
Identity-Based Networking Services commands	Cisco IOS Identity-Based Networking Services Command Reference

Standards and RFCs

Standard/RFC	Title
RFC 5176	Dynamic Authorization Extensions to RADIUS

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

Feature Information for RADIUS Change-of-Authorization Support

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Table 118: Feature Information for RADIUS Change-of-Authorization Support

Feature Name	Releases	Feature Information
RADIUS Change-of-Authorization	Cisco IOS Release 15.2(1)E	<p>Supports CoA requests for initiating the following:</p> <ul style="list-style-type: none">• Activating and deactivating service templates on sessions• Port bounce• Port shutdown• Querying a session• Reauthenticating a session• Terminating a session <p>These VSAs are sent in a standard CoA-Request message from a AAA server.</p>



CHAPTER 54

Local AAA Server

The Local AAA Server feature allows you to configure your device so that user authentication and authorization attributes currently available on AAA servers are available locally on the device. The attributes can be added to existing framework, such as the local user database or subscriber profile. The local AAA server provides access to the complete dictionary of Cisco IOS supported attributes.

- [Prerequisites for Local AAA Server, on page 1133](#)
- [Information About Local AAA Server, on page 1133](#)
- [How to Configure a Local AAA Server, on page 1135](#)
- [Configuration Examples for Local AAA Server, on page 1139](#)
- [Additional References for Local AAA Server, on page 1141](#)
- [Feature Information for Local AAA Server, on page 1141](#)

Prerequisites for Local AAA Server

- Before using this feature, enable the `aaa new-model` command.

Information About Local AAA Server

Local Authorization Attributes Overview

The AAA subsystem (authentication, authorization, and accounting) is responsible for managing all supported attributes that are available to the various services within the Cisco IOS software. As such, it maintains its own local dictionary of all supported attributes.

Local AAA Attribute Support

You can configure your device so that AAA authentication and authorization attributes currently available on AAA servers are made available on existing Cisco IOS devices. The attributes can be added to existing framework, such as the local user database or subscriber profile. For example, an attribute list can now be added to an existing username, providing the ability for the local user database to act as a local AAA server. For situations in which the local username list is relatively small, this flexibility allows you to provide complete user authentication or authorization locally within the Cisco IOS software without having a AAA server. This

ability can allow you to maintain your user database locally or provide a failover local mechanism without having to sacrifice policy options when defining local users.

A subscriber profile allows domain-based clients to have policy applied at the end-user service level. This flexibility allows common policy to be set for all users under a domain in one place and applied there whether or not user authorization is done locally. An attribute list can be added to the subscriber profile, allowing the profile to apply all attributes that can be applied to services using AAA servers. Attributes that are configured under the AAA attribute list are merged with the existing attributes that are generated with the existing subscriber profile and passed to the Subscriber Server Switch (SSS) framework for application.



Note Accounting is still done on a AAA server and is not supported by this feature.

AAA Attribute Lists

AAA attribute lists define user profiles that are local to the router. Every attribute that is known to the AAA subsystem is made available for configuration.

The AAA attributes that are defined in the AAA attribute list are standard RADIUS or TACACS+ attributes. However, they are in the Cisco IOS internal format for that attribute. The attributes must be converted from the RADIUS format (for a RADIUS case) to the Cisco IOS AAA interface format. TACACS+ attributes are generally identical to the Cisco IOS AAA interface format.

Converting from RADIUS Format to Cisco IOS AAA Format

You can use the **show aaa attributes protocol radius** command to get the Cisco IOS AAA format of the Internet Engineering Task Force (IETF) RADIUS attribute. The **show** command output provides a complete list of all the AAA attributes that are supported.



Note The conversion from RADIUS to internal AAA is done internally within the AAA framework. RADIUS vendor-specific attributes (VSAs) are usually accurately reflected during conversion. TACACS+ attributes are also usually identical to the local attributes and do not require the conversion process. However, IETF numbered attributes and some special VSAs often require the conversion process.

Validation of Attributes

Attributes are not validated at configuration. The AAA subsystem “knows” only the format that is expected by the services when the service defines a given attribute inside a definition file. However, it cannot validate the attribute information itself. This validation is done by a service when it first uses the attribute. This validation applies whether the AAA server is RADIUS or TACACS+. Thus, if you are not familiar with configuring a AAA server, it is advisable that you test your attribute list on a test device with the service that will be using the list before configuring and using it in a production environment.

How to Configure a Local AAA Server

Defining a AAA Attribute List

To define an AAA attribute list, perform the following steps.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **aaa attribute list** *list-name*
4. **attribute type** *{name}* *{value}* [**service** *service*] [**protocol** *protocol*]
5. **attribute type** *{name}* *{value}* [**service** *service*] [**protocol** *protocol*]
6. **attribute type** *{name}* *{value}* [**service** *service*] [**protocol** *protocol*]
7. **attribute type** *{name}* *{value}*
8. **attribute type** *{name}* *{value}*
9. **attribute type** *{name}* *{value}*
10. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	aaa attribute list <i>list-name</i> Example: Device(config)# aaa attribute list TEST	Defines a AAA attribute list and enters attribute list configuration mode.
Step 4	attribute type <i>{name}</i> <i>{value}</i> [service <i>service</i>] [protocol <i>protocol</i>] Example: Device(config-attr-list)# attribute type addr-pool poolname service ppp protocol ip	Defines an IP address pool to use.
Step 5	attribute type <i>{name}</i> <i>{value}</i> [service <i>service</i>] [protocol <i>protocol</i>]	Defines the loopback interface to use.

	Command or Action	Purpose
	Example: <pre>Device(config-attr-list)# attribute type ip-unnumbered loopbacknumber service ppp protocol ip</pre>	
Step 6	attribute type <i>{name}</i> <i>{value}</i> [service <i>service</i>] [protocol <i>protocol</i>] Example: <pre>Device(config-attr-list)# attribute type vrf-id vrfname service ppp protocol ip</pre>	Defines the virtual route forwarding (VRF) to use.
Step 7	attribute type <i>{name}</i> <i>{value}</i> Example: <pre>Device(config-attr-list)# attribute type ppp-authen-list aaalistname</pre>	Defines the AAA authentication list to use.
Step 8	attribute type <i>{name}</i> <i>{value}</i> Example: <pre>Device(config-attr-list)# attribute type ppp-author-list aaalistname</pre>	Defines the AAA authorization list to use.
Step 9	attribute type <i>{name}</i> <i>{value}</i> Example: <pre>Device(config-attr-list)# attribute type ppp-acct-list "aaa list name"</pre>	Defines the AAA accounting list to use.
Step 10	end Example: <pre>Device(config-attr-list)# end</pre>	Exits attribute list configuration mode and returns to privileged EXEC mode.

Defining a Subscriber Profile

To define a subscriber profile, perform the following steps.



Note RADIUS users should use the **show aaa attributes** command to map the RADIUS version of the particular attribute to the Cisco IOS AAA version of the string attribute. See the example Mapping from the RADIUS Version of a Particular Attribute to the Cisco IOS AAA Version Example.

SUMMARY STEPS

1. enable

2. **configure terminal**
3. **subscriber authorization enable**
4. **policy-map type service** *domain-name*
5. **service local**
6. **exit**
7. **aaa attribute list** *list-name*
8. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	subscriber authorization enable Example: Device(config)# subscriber authorization enable	Enables subscriber authorization.
Step 4	policy-map type service <i>domain-name</i> Example: Device(config)# policy-map type example.com	Specifies the username domain that has to be matched and enters subscriber profile configuration mode.
Step 5	service local Example: Device(subscriber-profile)# service local	Specifies that local subscriber authorization should be performed.
Step 6	exit Example: Device(subscriber-profile)# exit	Exits subscriber profile configuration mode.
Step 7	aaa attribute list <i>list-name</i> Example: Device(config)# aaa attribute list TEST	Defines the AAA attribute list from which RADIUS attributes are retrieved.
Step 8	end Example:	Exits global configuration mode and returns to privileged EXEC mode.

	Command or Action	Purpose
	Device(config)# end	

Monitoring and Troubleshooting a Local AAA Server

The following debug commands may be helpful in monitoring and troubleshooting, especially to ensure that domain-based service authorization is being triggered and that location authorization is being called on the local AAA server, which triggers the service.

SUMMARY STEPS

1. **enable**
2. **debug aaa authentication**
3. **debug aaa authorization**
4. **debug aaa per-user**
5. **debug ppp authentication**
6. **debug ppp error**
7. **debug ppp forward**
8. **debug ppp negotiation**
9. **debug radius**
10. **debug sss error**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	debug aaa authentication Example: Device# debug aaa authentication	Displays the methods of authentication being used and the results of these methods.
Step 3	debug aaa authorization Example: Device# debug aaa authorization	Displays the methods of authorization being used and the results of these methods.
Step 4	debug aaa per-user Example: Device# debug aaa per-user	Displays information about PPP session per-user activities.

	Command or Action	Purpose
Step 5	debug ppp authentication Example: Device# debug ppp authentication	Indicates whether a client is passing authentication.
Step 6	debug ppp error Example: Device (config)# debug ppp error	Displays protocol errors and error statistics that are associated with PPP connection negotiation and operation.
Step 7	debug ppp forward Example: Device# debug ppp forward	Displays who is taking control of a session.
Step 8	debug ppp negotiation Example: Device# debug ppp negotiation	Displays PPP packets sent during PPP startup, where PPP options are negotiated.
Step 9	debug radius Example: Device# debug radius	Displays information about the RADIUS server.
Step 10	debug sss error Example: Device# debug sss error	Displays diagnostic information about errors that may occur during SSS call setup.

Configuration Examples for Local AAA Server

Example: Local AAA Server

The following example shows a Point-to-Point over Ethernet (PPPoE) group named “bba-group” that is configured for subscriber profile cisco.com (thus, any user with the domain name cisco.com will execute the subscriber profile cisco.com authorization policy). The cisco.com subscriber profile is configured to attach the AAA attribute list “TEST,” which has both **ip vrf forwarding** and **ip unnumbered** commands configured for PPP service under Link Control Protocol (LCP) negotiation. This configuration will essentially cause the named attributes to be applied on the session with the cisco.com domain under the bba-group “pppoe grp1.”

```
aaa authentication ppp template1 local
aaa authorization network template1 local
!
aaa attribute list TEST
```

```

        attribute type interface-config "ip unnumbered FastEthernet0" service ppp protocol lcp
        attribute type interface-config "ip vrf forwarding blue" service ppp protocol lcp
    !
ip vrf blue
description vrf blue template1
rd 1:1
route-target export 1:1
route-target import 1:1
!
subscriber authorization enable
!
policy-map type service example.com
service local
aaa attribute list TEST
!
bba-group pppoe grp1
virtual-template 1
service profile example.com
!
interface Virtual-Template1
no ip address
no snmp trap link-status
no peer default ip address
no keepalive
ppp authentication pap template1
ppp authorization template1
!

```



Note In some versions of Cisco IOS software, it is better to use the explicit attribute instead of interface-config because it provides better scalability (full VAccess interfaces are not required, and subinterfaces could be used to provide the service). In such a case, you can configure **attribute type ip-unnumbered interface service ppp protocol ip** instead of **attribute type interface-config ip unnumbered interface service ppp protocol lcp**.

Example: Mapping from the RADIUS Version of a Particular Attribute to the Cisco IOS AAA Version

The following output example of the **show aaa attributes** command lists RADIUS attributes, which can be used when configuring this feature.

```

Device#
show aaa attributes protocol radius

IETF defined attributes:
  Type=4      Name=acl                      Format=Ulong
  Protocol:RADIUS
  Unknown    Type=11  Name=Filter-Id          Format=Binary
Converts attribute 11 (Filter-Id) of type Binary into an internal attribute
named "acl" of type Ulong. As such, one can configure this attributes locally
by using the attribute type "acl."
Cisco VSA attributes:
  Type=157   Name=interface-config      Format=String
Simply expects a string for the attribute of type "interface-config."

```



Note The **aaa attribute list** command requires the Cisco IOS AAA version of an attribute, which is defined in the “Name” field above.

Additional References for Local AAA Server

Related Documents

Related Topic	Document Title
Cisco security commands	<ul style="list-style-type: none"> • Cisco IOS Security Command Reference: Commands A to C • Cisco IOS Security Command Reference: Commands D to L • Cisco IOS Security Command Reference: Commands M to R • Cisco IOS Security Command Reference: Commands S to Z

Technical Assistance

Description	Link
<p>The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies.</p> <p>To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds.</p> <p>Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.</p>	http://www.cisco.com/cisco/web/support/index.html

Feature Information for Local AAA Server

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Table 119: Feature Information for Local AAA Server

Feature Name	Releases	Feature Information
Local AAA Server	Cisco IOS 15.2(1)E	The Local AAA Server feature allows you to configure your device so that user authentication and authorization attributes currently available on AAA servers are available locally on the device. The attributes can be added to existing framework, such as the local user database or subscriber profile. The local AAA server provides access to the complete dictionary of Cisco IOS supported attributes.



CHAPTER 55

Configuring Kerberos

- [Prerequisites for Controlling Switch Access with Kerberos, on page 1143](#)
- [Restrictions for Controlling Switch Access with Kerberos, on page 1143](#)
- [Information about Kerberos, on page 1144](#)
- [How to Configure Kerberos, on page 1147](#)
- [Monitoring the Kerberos Configuration, on page 1147](#)
- [Additional References, on page 1147](#)
- [Feature Information for Kerberos, on page 1148](#)

Prerequisites for Controlling Switch Access with Kerberos

The following are the prerequisites for controlling switch access with Kerberos.

- So that remote users can authenticate to network services, you must configure the hosts and the KDC in the Kerberos realm to communicate and mutually authenticate users and network services. To do this, you must identify them to each other. You add entries for the hosts to the Kerberos database on the KDC and add KEYTAB files generated by the KDC to all hosts in the Kerberos realm. You also create entries for the users in the KDC database.
- A Kerberos server can be a switch that is configured as a network security server and that can authenticate users by using the Kerberos protocol.

When you add or create entries for the hosts and users, follow these guidelines:

- The Kerberos principal name *must* be in all lowercase characters.
- The Kerberos instance name *must* be in all lowercase characters.
- The Kerberos realm name *must* be in all uppercase characters.

Restrictions for Controlling Switch Access with Kerberos

The following lists any restrictions for controlling switch access with Kerberos.

Information about Kerberos

This section provides Kerberos information.

Kerberos and Switch Access

This section describes how to enable and configure the Kerberos security system, which authenticates requests for network resources by using a trusted third party.



Note In the Kerberos configuration examples, the trusted third party can be any switch that supports Kerberos, that is configured as a network security server, and that can authenticate users by using the Kerberos protocol.

Kerberos Overview

Kerberos is a secret-key network authentication protocol, which was developed at the Massachusetts Institute of Technology (MIT). It uses the Data Encryption Standard (DES) cryptographic algorithm for encryption and authentication and authenticates requests for network resources. Kerberos uses the concept of a trusted third party to perform secure verification of users and services. This trusted third party is called the *key distribution center* (KDC).

Kerberos verifies that users are who they claim to be and the network services that they use are what the services claim to be. To do this, a KDC or trusted Kerberos server issues tickets to users. These tickets, which have a limited life span, are stored in user credential caches. The Kerberos server uses the tickets instead of user names and passwords to authenticate users and network services.



Note A Kerberos server can be any switch that is configured as a network security server and that can authenticate users by using the Kerberos protocol.

The Kerberos credential scheme uses a process called *single logon*. This process authenticates a user once and then allows secure authentication (without encrypting another password) wherever that user credential is accepted.

This software release supports Kerberos 5, which allows organizations that are already using Kerberos 5 to use the same Kerberos authentication database on the KDC that they are already using on their other network hosts (such as UNIX servers and PCs).

Kerberos supports these network services:

- Telnet
- rlogin
- rsh

This table lists the common Kerberos-related terms and definitions.

Table 120: Kerberos Terms

Term	Definition
Authentication	A process by which a user or service identifies itself to another service. For example, a client can authenticate to a switch or a switch can authenticate to another switch.
Authorization	A means by which the switch identifies what privileges the user has in a network or on the switch and what actions the user can perform.
Credential	A general term that refers to authentication tickets, such as TGTs ⁸ and service credentials. Kerberos credentials verify the identity of a user or service. If a network service decides to trust the Kerberos server that issued a ticket, it can be used in place of re-entering a username and password. Credentials have a default life span of eight hours.
Instance	An authorization level label for Kerberos principals. Most Kerberos principals are of the form <i>user@REALM</i> (for example, <i>smith@EXAMPLE.COM</i>). A Kerberos principal with a Kerberos instance has the form <i>user/instance@REALM</i> (for example, <i>smith/admin@EXAMPLE.COM</i>). The Kerberos instance can be used to specify the authorization level for the user if authentication is successful. The server of each network service might implement and enforce the authorization mappings of Kerberos instances but is not required to do so. Note The Kerberos principal and instance names <i>must</i> be in all lowercase characters. Note The Kerberos realm name <i>must</i> be in all uppercase characters.
KDC ⁹	Key distribution center that consists of a Kerberos server and database program that is running on a network host.
Kerberized	A term that describes applications and services that have been modified to support the Kerberos credential infrastructure.
Kerberos realm	A domain consisting of users, hosts, and network services that are registered to a Kerberos server. The Kerberos server is trusted to verify the identity of a user or network service to another user or network service. Note The Kerberos realm name <i>must</i> be in all uppercase characters.
Kerberos server	A daemon that is running on a network host. Users and network services register their identity with the Kerberos server. Network services query the Kerberos server to authenticate to other network services.
KEYTAB ¹⁰	A password that a network service shares with the KDC. In Kerberos 5 and later Kerberos versions, the network service authenticates an encrypted service credential by using the KEYTAB to decrypt it. In Kerberos versions earlier than Kerberos 5, KEYTAB is referred to as SRVTAB ¹¹ .
Principal	Also known as a Kerberos identity, this is who you are or what a service is according to the Kerberos server. Note The Kerberos principal name <i>must</i> be in all lowercase characters.

Term	Definition
Service credential	A credential for a network service. When issued from the KDC, this credential is encrypted with the password shared by the network service and the KDC. The password is also shared with the user TGT.
SRVTAB	A password that a network service shares with the KDC. In Kerberos 5 or later Kerberos versions, SRVTAB is referred to as KEYTAB.
TGT	Ticket granting ticket that is a credential that the KDC issues to authenticated users. When users receive a TGT, they can authenticate to network services within the Kerberos realm represented by the KDC.

- ⁸ ticket granting ticket
- ⁹ key distribution center
- ¹⁰ key table
- ¹¹ server table

Kerberos Operation

A Kerberos server can be a device that is configured as a network security server and that can authenticate remote users by using the Kerberos protocol. Although you can customize Kerberos in a number of ways, remote users attempting to access network services must pass through three layers of security before they can access network services.

To authenticate to network services by using a device as a Kerberos server, remote users must follow these steps:

Authenticating to a Boundary Switch

This section describes the first layer of security through which a remote user must pass. The user must first authenticate to the boundary switch. This process then occurs:

1. The user opens an un-Kerberized Telnet connection to the boundary switch.
2. The switch prompts the user for a username and password.
3. The switch requests a TGT from the KDC for this user.
4. The KDC sends an encrypted TGT that includes the user identity to the switch.
5. The switch attempts to decrypt the TGT by using the password that the user entered.
 - If the decryption is successful, the user is authenticated to the switch.
 - If the decryption is not successful, the user repeats Step 2 either by re-entering the username and password (noting if Caps Lock or Num Lock is on or off) or by entering a different username and password.

A remote user who initiates a un-Kerberized Telnet session and authenticates to a boundary switch is inside the firewall, but the user must still authenticate directly to the KDC before getting access to the network services. The user must authenticate to the KDC because the TGT that the KDC issues is stored on the switch and cannot be used for additional authentication until the user logs on to the switch.

Obtaining a TGT from a KDC

This section describes the second layer of security through which a remote user must pass. The user must now authenticate to a KDC and obtain a TGT from the KDC to access network services.

For instructions about how to authenticate to a KDC, see the “Obtaining a TGT from a KDC” section in the “Security Server Protocols” chapter of the *Cisco IOS Security Configuration Guide, Release 12.4*.

Authenticating to Network Services

This section describes the third layer of security through which a remote user must pass. The user with a TGT must now authenticate to the network services in a Kerberos realm.

For instructions about how to authenticate to a network service, see the “Authenticating to Network Services” section in the “Security Server Protocols” chapter of the *Cisco IOS Security Configuration Guide, Release 12.4*.

How to Configure Kerberos

To set up a Kerberos-authenticated server-client system, follow these steps:

- Configure the KDC by using Kerberos commands.
- Configure the switch to use the Kerberos protocol.

Monitoring the Kerberos Configuration

To display the Kerberos configuration, use the following commands:

- **show running-config**
- **show kerberos creds**: Lists the credentials in a current user’s credentials cache.
- **clear kerberos creds**: Destroys all credentials in a current user’s credentials cache, including those forwarded.

Additional References

Related Documents

Related Topic	Document Title
Kerberos Commands	<i>Cisco IOS Security Command Reference</i>

Error Message Decoder

Description	Link
To help you research and resolve system error messages in this release, use the Error Message Decoder tool.	https://www.cisco.com/cgi-bin/Support/Errordecoder/index.cgi

MIBs

MIB	MIBs Link
All the supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
<p>The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies.</p> <p>To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds.</p> <p>Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.</p>	http://www.cisco.com/support

Feature Information for Kerberos

Release	Feature Information
Cisco IOS Release 15.0(2)EX1	This feature was introduced.



CHAPTER 56

Configuring Accounting

The AAA Accounting feature allows the services that users are accessing and the amount of network resources that users are consuming to be tracked. When AAA Accounting is enabled, the network access server reports user activity to the TACACS+ or RADIUS security server (depending on which security method is implemented) in the form of accounting records. Each accounting record contains accounting attribute-value (AV) pairs and is stored on the security server. This data can then be analyzed for network management, client billing, and auditing.

- [Prerequisites for Configuring Accounting, on page 1149](#)
- [Restrictions for Configuring Accounting, on page 1149](#)
- [Information About Configuring Accounting, on page 1150](#)
- [How to Configure Accounting, on page 1163](#)
- [Configuration Examples for Accounting, on page 1172](#)
- [Additional References for Configuring Accounting, on page 1176](#)
- [Feature Information for Configuring Accounting, on page 1176](#)

Prerequisites for Configuring Accounting

The following tasks must be performed before configuring accounting using named method lists:

- Enable AAA on the network access server by using the **aaa new-model** command in global configuration mode.
- Define the characteristics of the RADIUS or TACACS+ security server if RADIUS or TACACS+ authorization is issued. For more information about configuring the Cisco network access server to communicate with the RADIUS security server, see the Configuring RADIUS module. For more information about configuring the Cisco network access server to communicate with the TACACS+ security server, see the Configuring TACACS+ module.

Restrictions for Configuring Accounting

- Accounting information can be sent simultaneously to a maximum of only four AAA servers.
- For Service Selection Gateway (SSG) systems, the **aaa accounting network broadcast** command broadcasts only **start-stop** accounting records. If interim accounting records are configured using the

ssg accounting interval command, the interim accounting records are sent only to the configured default RADIUS server.

Information About Configuring Accounting

Named Method Lists for Accounting

Similar to authentication and authorization method lists, method lists for accounting define the way accounting is performed and the sequence in which these methods are performed.

Named accounting method lists allow particular security protocol to be designated and used on specific lines or interfaces for accounting services. The only exception is the default method list (which is named “default”). The default method list is automatically applied to all interfaces except those that have a named method list explicitly defined. A defined method list overrides the default method list.

A method list is simply a named list describing the accounting methods to be queried (such as RADIUS or TACACS+), in sequence. Method lists allow one or more security protocols to be designated and used for accounting, thus ensuring a backup system for accounting in case the initial method fails. Cisco IOS software uses the first method listed to support accounting; if that method fails to respond, the Cisco IOS software selects the next accounting method listed in the method list. This process continues until there is successful communication with a listed accounting method, or all methods defined are exhausted.



Note The Cisco IOS software attempts accounting with the next listed accounting method only when there is no response from the previous method. If accounting fails at any point in this cycle--meaning that the security server responds by denying the user access--the accounting process stops and no other accounting methods are attempted.

Accounting method lists are specific to the type of accounting being requested. AAA supports seven different types of accounting:

- **Network** --Provides information for all PPP, SLIP, or ARAP sessions, including packet and byte counts.
- **EXEC** --Provides information about user EXEC terminal sessions of the network access server.
- **Commands** --Provides information about the EXEC mode commands that a user issues. Command accounting generates accounting records for all EXEC mode commands, including global configuration commands, associated with a specific privilege level.
- **Connection** --Provides information about all outbound connections made from the network access server, such as Telnet, local-area transport (LAT), TN3270, packet assembler/disassembler (PAD), and rlogin.
- **System** --Provides information about system-level events.
- **Resource** --Provides “start” and “stop” records for calls that have passed user authentication, and provides “stop” records for calls that fail to authenticate.
- **VRRS** --Provides information about Virtual Router Redundancy Service (VRRS).



Note System accounting does not use named accounting lists; only the default list for system accounting can be defined.

Once again, when a named method list is created, a particular list of accounting methods for the indicated accounting type are defined.

Accounting method lists must be applied to specific lines or interfaces before any of the defined methods are performed. The only exception is the default method list (which is named “default”). If the **aaa accounting** command for a particular accounting type is issued without specifying a named method list, the default method list is automatically applied to all interfaces or lines except those that have a named method list explicitly defined (A defined method list overrides the default method list). If no default method list is defined, then no accounting takes place.

This section includes the following subsections:

Method Lists and Server Groups

A server group is a way to group existing LDAP, RADIUS, or TACACS+ server hosts for use in method lists. The figure below shows a typical AAA network configuration that includes four security servers: R1 and R2 are RADIUS servers, and T1 and T2 are TACACS+ servers. R1 and R2 make up the group of RADIUS servers. T1 and T2 make up the group of TACACS+ servers.

Using server groups, a subset of the configured server hosts can be specified and use them for a particular service. For example, server groups allows R1 and R2 to be defined as separate server groups, and T1 and T2 as separate server groups. This allows either R1 and T1 to be specified in the method list or R2 and T2 in the method list, which provides more flexibility in the way that RADIUS and TACACS+ resources are assigned.

Server groups also can include multiple host entries for the same server, as long as each entry has a unique identifier. The combination of an IP address and a UDP port number creates a unique identifier, allowing different ports to be individually defined as RADIUS hosts providing a specific AAA service. In other words, this unique identifier enables RADIUS requests to be sent to different UDP ports on a server at the same IP address. If two different host entries on the same RADIUS server are configured for the same service--for example, authorization--the second host entry configured acts as fail-over backup to the first one. Using this example, if the first host entry fails to provide accounting services, the network access server tries the second host entry configured on the same device for accounting services. (The RADIUS host entries are tried in the order they are configured.)

AAA Accounting Methods

The Cisco IOS software supports the following two methods for accounting:

- TACACS+--The network access server reports user activity to the TACACS+ security server in the form of accounting records. Each accounting record contains accounting AV pairs and is stored on the security server.
- RADIUS--The network access server reports user activity to the RADIUS security server in the form of accounting records. Each accounting record contains accounting AV pairs and is stored on the security server.



Note With CSCuc32663, passwords and accounting logs are masked before being sent to the TACACS+ or RADIUS security servers. Use the **aaa accounting commands visible-keys** command to send unmasked information to the TACACS+ or RADIUS security servers.

Accounting Record Types

For minimal accounting, use the **stop-only** keyword, which instructs the specified method (**RADIUS** or **TACACS+**) to send a stop record accounting notice at the end of the requested user process. For more accounting information, use the **start-stop** keyword to send a start accounting notice at the beginning of the requested event and a stop accounting notice at the end of the event. To stop all accounting activities on this line or interface, use the **none** keyword.

AAA Accounting Methods

The Cisco IOS software supports the following two methods for accounting:

- **TACACS+**--The network access server reports user activity to the TACACS+ security server in the form of accounting records. Each accounting record contains accounting AV pairs and is stored on the security server.
- **RADIUS**--The network access server reports user activity to the RADIUS security server in the form of accounting records. Each accounting record contains accounting AV pairs and is stored on the security server.



Note With CSCuc32663, passwords and accounting logs are masked before being sent to the TACACS+ or RADIUS security servers. Use the **aaa accounting commands visible-keys** command to send unmasked information to the TACACS+ or RADIUS security servers.

AAA Accounting Types

Network Accounting

Network accounting provides information for all PPP, SLIP, or ARAP sessions, including packet and byte counts.

The following example shows the information contained in a RADIUS network accounting record for a PPP user who comes in through an EXEC session:

```
Wed Jun 27 04:44:45 2001
  NAS-IP-Address = "172.16.25.15"
  NAS-Port = 5
  User-Name = "username1"
  Client-Port-DNIS = "4327528"
  Caller-ID = "562"
  Acct-Status-Type = Start
  Acct-Authentic = RADIUS
  Service-Type = Exec-User
  Acct-Session-Id = "0000000D"
```

```

Acct-Delay-Time = 0
User-Id = "username1"
NAS-Identifier = "172.16.25.15"

Wed Jun 27 04:45:00 2001
NAS-IP-Address = "172.16.25.15"
NAS-Port = 5
User-Name = "username1"
Client-Port-DNIS = "4327528"
Caller-ID = "562"
Acct-Status-Type = Start
Acct-Authentic = RADIUS
Service-Type = Framed
Acct-Session-Id = "0000000E"
Framed-IP-Address = "10.1.1.2"
Framed-Protocol = PPP
Acct-Delay-Time = 0
User-Id = "username1"
NAS-Identifier = "172.16.25.15"

Wed Jun 27 04:47:46 2001
NAS-IP-Address = "172.16.25.15"
NAS-Port = 5
User-Name = "username1"
Client-Port-DNIS = "4327528"
Caller-ID = "562"
Acct-Status-Type = Stop
Acct-Authentic = RADIUS
Service-Type = Framed
Acct-Session-Id = "0000000E"
Framed-IP-Address = "10.1.1.2"
Framed-Protocol = PPP
Acct-Input-Octets = 3075
Acct-Output-Octets = 167
Acct-Input-Packets = 39
Acct-Output-Packets = 9
Acct-Session-Time = 171
Acct-Delay-Time = 0
User-Id = "username1"
NAS-Identifier = "172.16.25.15"

Wed Jun 27 04:48:45 2001
NAS-IP-Address = "172.16.25.15"
NAS-Port = 5
User-Name = "username1"
Client-Port-DNIS = "4327528"
Caller-ID = "408"
Acct-Status-Type = Stop
Acct-Authentic = RADIUS
Service-Type = Exec-User
Acct-Session-Id = "0000000D"
Acct-Delay-Time = 0
User-Id = "username1"
NAS-Identifier = "172.16.25.15"

```

The following example shows the information contained in a TACACS+ network accounting record for a PPP user who first started an EXEC session:

```

Wed Jun 27 04:00:35 2001 172.16.25.15  username1  tty4  562/4327528  starttask_id=28
      service=shell
Wed Jun 27 04:00:46 2001 172.16.25.15  username1  tty4  562/4327528  starttask_id=30
      addr=10.1.1.1  service=ppp
Wed Jun 27 04:00:49 2001 172.16.25.15  username1  tty4  408/4327528  update
task_id=30  addr=10.1.1.1  service=ppp  protocol=ip  addr=10.1.1.1
Wed Jun 27 04:01:31 2001 172.16.25.15  username1  tty4  562/4327528  stoptask_id=30
      addr=10.1.1.1  service=ppp  protocol=ip  addr=10.1.1.1  bytes_in=2844

```

```

bytes_out=1682 paks_in=36 paks_out=24 elapsed_time=51
Wed Jun 27 04:01:32 2001 172.16.25.15 username1 tty4 562/4327528 stoptask_id=28
service=shell elapsed_time=57

```



Note The precise format of accounting packets records may vary depending on the security server daemon.

The following example shows the information contained in a RADIUS network accounting record for a PPP user who comes in through autoselect:

```

Wed Jun 27 04:30:52 2001
NAS-IP-Address = "172.16.25.15"
NAS-Port = 3
User-Name = "username1"
Client-Port-DNIS = "4327528"
Caller-ID = "562"
Acct-Status-Type = Start
Acct-Authentic = RADIUS
Service-Type = Framed
Acct-Session-Id = "0000000B"
Framed-Protocol = PPP
Acct-Delay-Time = 0
User-Id = "username1"
NAS-Identifier = "172.16.25.15"

```

```

Wed Jun 27 04:36:49 2001
NAS-IP-Address = "172.16.25.15"
NAS-Port = 3
User-Name = "username1"
Client-Port-DNIS = "4327528"
Caller-ID = "562"
Acct-Status-Type = Stop
Acct-Authentic = RADIUS
Service-Type = Framed
Acct-Session-Id = "0000000B"
Framed-Protocol = PPP
Framed-IP-Address = "10.1.1.1"
Acct-Input-Octets = 8630
Acct-Output-Octets = 5722
Acct-Input-Packets = 94
Acct-Output-Packets = 64
Acct-Session-Time = 357
Acct-Delay-Time = 0
User-Id = "username1"
NAS-Identifier = "172.16.25.15"

```

The following example shows the information contained in a TACACS+ network accounting record for a PPP user who comes in through autoselect:

```

Wed Jun 27 04:02:19 2001 172.16.25.15 username1 Async5 562/4327528 starttask_id=35
service=ppp
Wed Jun 27 04:02:25 2001 172.16.25.15 username1 Async5 562/4327528 update
task_id=35 service=ppp protocol=ip addr=10.1.1.2
Wed Jun 27 04:05:03 2001 172.16.25.15 username1 Async5 562/4327528 stoptask_id=35
service=ppp protocol=ip addr=10.1.1.2 bytes_in=3366 bytes_out=2149
paks_in=42 paks_out=28 elapsed_time=164

```

EXEC Accounting

EXEC accounting provides information about user EXEC terminal sessions (user shells) on the network access server, including username, date, start and stop times, the access server IP address, and (for dial-in users) the telephone number the call originated from.

The following example shows the information contained in a RADIUS EXEC accounting record for a dial-in user:

```

Wed Jun 27 04:26:23 2001
  NAS-IP-Address = "172.16.25.15"
  NAS-Port = 1
  User-Name = "username1"
  Client-Port-DNIS = "4327528"
  Caller-ID = "5622329483"
  Acct-Status-Type = Start
  Acct-Authentic = RADIUS
  Service-Type = Exec-User
  Acct-Session-Id = "00000006"
  Acct-Delay-Time = 0
  User-Id = "username1"
  NAS-Identifier = "172.16.25.15"
Wed Jun 27 04:27:25 2001
  NAS-IP-Address = "172.16.25.15"
  NAS-Port = 1
  User-Name = "username1"
  Client-Port-DNIS = "4327528"
  Caller-ID = "5622329483"
  Acct-Status-Type = Stop
  Acct-Authentic = RADIUS
  Service-Type = Exec-User
  Acct-Session-Id = "00000006"
  Acct-Session-Time = 62
  Acct-Delay-Time = 0
  User-Id = "username1"
  NAS-Identifier = "172.16.25.15"

```

The following example shows the information contained in a TACACS+ EXEC accounting record for a dial-in user:

```

Wed Jun 27 03:46:21 2001      172.16.25.15      username1      tty3      5622329430/4327528
start      task_id=2      service=shell
Wed Jun 27 04:08:55 2001      172.16.25.15      username1      tty3      5622329430/4327528
stop      task_id=2      service=shell      elapsed_time=1354

```

The following example shows the information contained in a RADIUS EXEC accounting record for a Telnet user:

```

Wed Jun 27 04:48:32 2001
  NAS-IP-Address = "172.16.25.15"
  NAS-Port = 26
  User-Name = "username1"
  Caller-ID = "10.68.202.158"
  Acct-Status-Type = Start
  Acct-Authentic = RADIUS
  Service-Type = Exec-User
  Acct-Session-Id = "00000010"
  Acct-Delay-Time = 0
  User-Id = "username1"
  NAS-Identifier = "172.16.25.15"

```

```

Wed Jun 27 04:48:46 2001
NAS-IP-Address = "172.16.25.15"
NAS-Port = 26
User-Name = "username1"
Caller-ID = "10.68.202.158"
Acct-Status-Type = Stop
Acct-Authentic = RADIUS
Service-Type = Exec-User
Acct-Session-Id = "00000010"
Acct-Session-Time = 14
Acct-Delay-Time = 0
User-Id = "username1"
NAS-Identifier = "172.16.25.15"

```

The following example shows the information contained in a TACACS+ EXEC accounting record for a Telnet user:

```

Wed Jun 27 04:06:53 2001      172.16.25.15      username1      tty26      10.68.202.158
starttask_id=41      service=shell
Wed Jun 27 04:07:02 2001      172.16.25.15      username1      tty26      10.68.202.158
stoptask_id=41      service=shell      elapsed_time=9

```

Command Accounting

Command accounting provides information about the EXEC shell commands for a specified privilege level that are being executed on a network access server. Each command accounting record includes a list of the commands executed for that privilege level, as well as the date and time each command was executed, and the user who executed it.

The following example shows the information contained in a TACACS+ command accounting record for privilege level 1:

```

Wed Jun 27 03:46:47 2001      172.16.25.15      username1      tty3      5622329430/4327528
stop      task_id=3      service=shell      priv-lvl=1      cmd=show version <cr>
Wed Jun 27 03:46:58 2001      172.16.25.15      username1      tty3      5622329430/4327528
stop      task_id=4      service=shell      priv-lvl=1      cmd=show interfaces Ethernet 0
<cr>
Wed Jun 27 03:47:03 2001      172.16.25.15      username1      tty3      5622329430/4327528
stop      task_id=5      service=shell      priv-lvl=1      cmd=show ip route <cr>

```

The following example shows the information contained in a TACACS+ command accounting record for privilege level 15:

```

Wed Jun 27 03:47:17 2001      172.16.25.15      username1      tty3      5622329430/4327528
stop      task_id=6      service=shell      priv-lvl=15      cmd=configure terminal <cr>
Wed Jun 27 03:47:21 2001      172.16.25.15      username1      tty3      5622329430/4327528
stop      task_id=7      service=shell      priv-lvl=15      cmd=interface Serial 0 <cr>
Wed Jun 27 03:47:29 2001      172.16.25.15      username1      tty3      5622329430/4327528
stop      task_id=8      service=shell      priv-lvl=15      cmd=ip address 10.1.1.1 255.255.255.0
<cr>

```



Note The Cisco implementation of RADIUS does not support command accounting.

Connection Accounting

Connection accounting provides information about all outbound connections made from the network access server such as Telnet, LAT, TN3270, PAD, and rlogin.

The following example shows the information contained in a RADIUS connection accounting record for an outbound Telnet connection:

```
Wed Jun 27 04:28:00 2001
  NAS-IP-Address = "172.16.25.15"
  NAS-Port = 2
  User-Name = "username1"
  Client-Port-DNIS = "4327528"
  Caller-ID = "5622329477"
  Acct-Status-Type = Start
  Acct-Authentic = RADIUS
  Service-Type = Login
  Acct-Session-Id = "00000008"
  Login-Service = Telnet
  Login-IP-Host = "10.68.202.158"
  Acct-Delay-Time = 0
  User-Id = "username1"
  NAS-Identifier = "172.16.25.15"
```

```
Wed Jun 27 04:28:39 2001
  NAS-IP-Address = "172.16.25.15"
  NAS-Port = 2
  User-Name = "username1"
  Client-Port-DNIS = "4327528"
  Caller-ID = "5622329477"
  Acct-Status-Type = Stop
  Acct-Authentic = RADIUS
  Service-Type = Login
  Acct-Session-Id = "00000008"
  Login-Service = Telnet
  Login-IP-Host = "10.68.202.158"
  Acct-Input-Octets = 10774
  Acct-Output-Octets = 112
  Acct-Input-Packets = 91
  Acct-Output-Packets = 99
  Acct-Session-Time = 39
  Acct-Delay-Time = 0
  User-Id = "username1"
  NAS-Identifier = "172.16.25.15"
```

The following example shows the information contained in a TACACS+ connection accounting record for an outbound Telnet connection:

```
Wed Jun 27 03:47:43 2001      172.16.25.15      username1  tty3      5622329430/4327528
start  task_id=10      service=connection      protocol=telnet  addr=10.68.202.158  cmd=telnet
      username1-sun
Wed Jun 27 03:48:38 2001      172.16.25.15      username1  tty3      5622329430/4327528
stop   task_id=10      service=connection      protocol=telnet  addr=10.68.202.158  cmd=telnet
      username1-sun      bytes_in=4467  bytes_out=96  paks_in=61      paks_out=72  elapsed_time=55
```

The following example shows the information contained in a RADIUS connection accounting record for an outbound rlogin connection:

```
Wed Jun 27 04:29:48 2001
  NAS-IP-Address = "172.16.25.15"
```

```

NAS-Port = 2
User-Name = "username1"
Client-Port-DNIS = "4327528"
Caller-ID = "5622329477"
Acct-Status-Type = Start
Acct-Authentic = RADIUS
Service-Type = Login
Acct-Session-Id = "0000000A"
Login-Service = Rlogin
Login-IP-Host = "10.68.202.158"
Acct-Delay-Time = 0
User-Id = "username1"
NAS-Identifier = "172.16.25.15"

Wed Jun 27 04:30:09 2001
NAS-IP-Address = "172.16.25.15"
NAS-Port = 2
User-Name = "username1"
Client-Port-DNIS = "4327528"
Caller-ID = "5622329477"
Acct-Status-Type = Stop
Acct-Authentic = RADIUS
Service-Type = Login
Acct-Session-Id = "0000000A"
Login-Service = Rlogin
Login-IP-Host = "10.68.202.158"
Acct-Input-Octets = 18686
Acct-Output-Octets = 86
Acct-Input-Packets = 90
Acct-Output-Packets = 68
Acct-Session-Time = 22
Acct-Delay-Time = 0
User-Id = "username1"
NAS-Identifier = "172.16.25.15"

```

The following example shows the information contained in a TACACS+ connection accounting record for an outbound rlogin connection:

```

Wed Jun 27 03:48:46 2001      172.16.25.15      username1      tty3      5622329430/4327528
start      task_id=12      service=connection      protocol=rlogin      addr=10.68.202.158      cmd=rlogin
username1-sun /user username1
Wed Jun 27 03:51:37 2001      172.16.25.15      username1      tty3      5622329430/4327528
stop      task_id=12      service=connection      protocol=rlogin      addr=10.68.202.158      cmd=rlogin
username1-sun /user username1      bytes_in=659926      bytes_out=138      paks_in=2378      paks_
out=1251      elapsed_time=171

```

The following example shows the information contained in a TACACS+ connection accounting record for an outbound LAT connection:

```

Wed Jun 27 03:53:06 2001      172.16.25.15      username1      tty3      5622329430/4327528
start      task_id=18      service=connection      protocol=lat      addr=VAX      cmd=lat
VAX
Wed Jun 27 03:54:15 2001      172.16.25.15      username1      tty3      5622329430/4327528
stop      task_id=18      service=connection      protocol=lat      addr=VAX      cmd=lat
VAX      bytes_in=0      bytes_out=0      paks_in=0      paks_out=0      elapsed_time=6

```

System Accounting

System accounting provides information about all system-level events (for example, when the system reboots or when accounting is turned on or off).

The following accounting record shows a typical TACACS+ system accounting record server indicating that AAA Accounting has been turned off:

```
Wed Jun 27 03:55:32 2001      172.16.25.15      unknown unknown unknown start   task_id=25
service=system event=sys_acct reason=reconfigure
```



Note The precise format of accounting packets records may vary depending on the TACACS+ daemon.

The following accounting record shows a TACACS+ system accounting record indicating that AAA Accounting has been turned on:

```
Wed Jun 27 03:55:22 2001      172.16.25.15      unknown unknown unknown stop    task_id=23
service=system event=sys_acct reason=reconfigure
```

Additional tasks for measuring system resources are covered in the Cisco IOS software configuration guides. For example, IP accounting tasks are described in the Configuring IP Services chapter in the *CiscoIOS Application Services Configuration Guide*.

Resource Accounting

The Cisco implementation of AAA accounting provides “start” and “stop” record support for calls that have passed user authentication. The additional feature of generating “stop” records for calls that fail to authenticate as part of user authentication is also supported. Such records are necessary for users employing accounting records to manage and monitor their networks.

This section includes the following subsections:

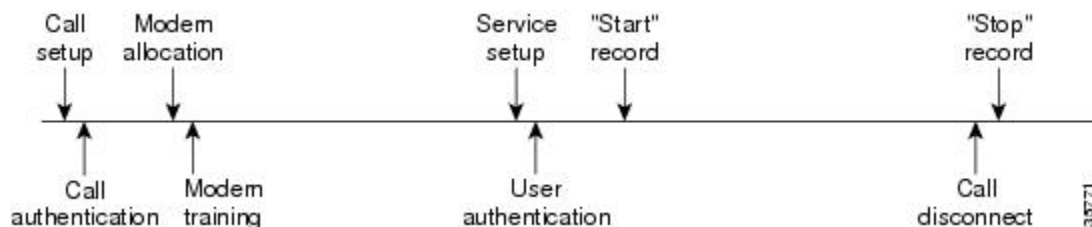
AAA Resource Failure Stop Accounting

Before AAA resource failure stop accounting, there was no method of providing accounting records for calls that failed to reach the user authentication stage of a call setup sequence. Such records are necessary for users employing accounting records to manage and monitor their networks and their wholesale customers.

This functionality generates a “stop” accounting record for any calls that do not reach user authentication; “stop” records are generated from the moment of call setup. All calls that pass user authentication behave as they did before; that is, no additional accounting records are seen.

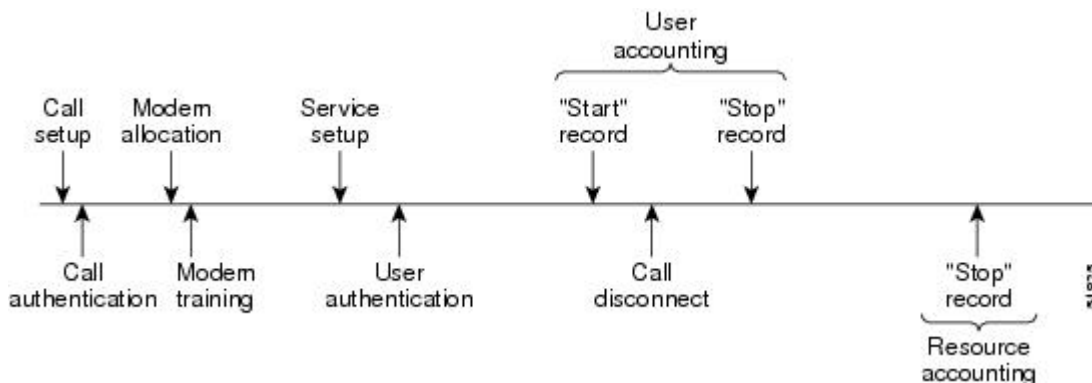
The figure below illustrates a call setup sequence with normal call flow (no disconnect) and without AAA resource failure stop accounting enabled.

Figure 88: Modem Dial-In Call Setup Sequence With Normal Flow and Without Resource Failure Stop Accounting Enabled



The figure below illustrates a call setup sequence with normal call flow (no disconnect) and with AAA resource failure stop accounting enabled.

Figure 89: Modem Dial-In Call Setup Sequence With Normal Flow and With Resource Failure Stop Accounting Enabled



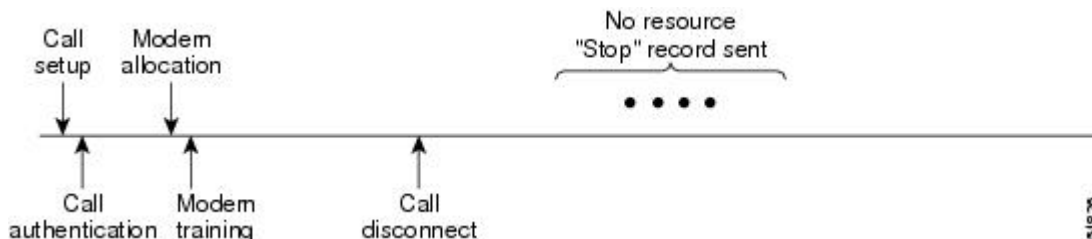
The figure below illustrates a call setup sequence with call disconnect occurring before user authentication and with AAA resource failure stop accounting enabled.

Figure 90: Modem Dial-In Call Setup Sequence With Call Disconnect Occurring Before User Authentication and With Resource Failure Stop Accounting Enabled



The figure below illustrates a call setup sequence with call disconnect occurring before user authentication and without AAA resource failure stop accounting enabled.

Figure 91: Modem Dial-In Call Setup Sequence With Call Disconnect Occurring Before User Authentication and Without Resource Failure Stop Accounting Enabled



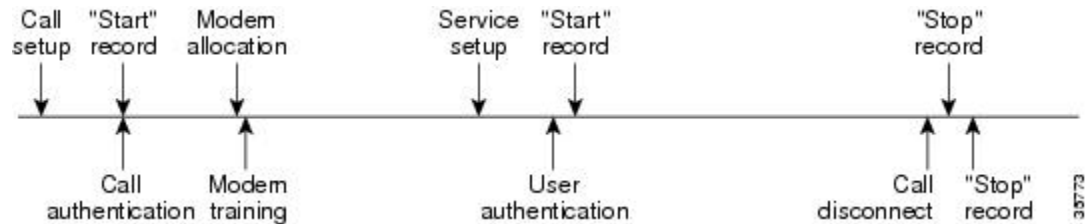
AAA Resource Accounting for Start-Stop Records

AAA resource accounting for start-stop records supports the ability to send a “start” record at each call setup, followed by a corresponding “stop” record at the call disconnect. This functionality can be used to manage and monitor wholesale customers from one source of data reporting, such as accounting records.

With this feature, a call setup and call disconnect “start-stop” accounting record tracks the progress of the resource connection to the device. A separate user authentication “start-stop” accounting record tracks the user management progress. These two sets of accounting records are interlinked by using a unique session ID for the call.

The figure below illustrates a call setup sequence with AAA resource start-stop accounting enabled.

Figure 92: Modem Dial-In Call Setup Sequence With Resource Start-Stop Accounting Enabled



VRRS Accounting

Virtual Router Redundancy Service (VRRS) provides a multiclient information abstraction and management service between a First Hop Redundancy Protocol (FHRP) and a registered client. The VRRS multiclient service provides a consistent interface with FHRP protocols by abstracting over several FHRPs and providing an idealized view of their state. VRRS manages data updates, allowing interested clients to register in one place and receive updates for named FHRP groups or all registered FHRP groups.

Virtual Router Redundancy Protocol (VRRP) is an FHRP that acts as a server that pushes FHRP status information out to all registered VRRS clients. Clients obtain status on essential information provided by the FHRP, including current and previous redundancy states, active and inactive L3 and L2 addresses, and, in some cases, information about other redundant gateways in the network. Clients can use this information to provide stateless and stateful redundancy information to clients and protocols.

VRRS Accounting Plug-in

The VRRS Accounting plug-in provides a configurable AAA method list mechanism that provides updates to a RADIUS server when a VRRS group transitions its state. The VRRS accounting plug-in is an extension of existing AAA system accounting messages. The VRRS Accounting plug-in provides accounting-on and accounting-off messages and an additional Vendor-Specific Attribute (VSA) that sends the configured VRRS name in RADIUS accounting messages. The VRRS name is configured using the **vrrp name** command in interface configuration mode.

The VRRS Accounting plug-in provides a configurable AAA method list mechanism that provides updates to a RADIUS server when a VRRS group transitions its state.

The VRRS accounting plug-in is an extension of existing AAA system accounting messages. The VRRS Accounting plug-in provides accounting-on and accounting-off messages and an additional Vendor-Specific Attribute (VSA) that sends the configured VRRS name in RADIUS accounting messages. The VRRS name is configured using the **vrrp name** command in interface configuration mode. The VRRS Accounting plug-in sends an accounting-on message to RADIUS when a VRRS group transitions to the active state, and it sends an accounting-off message when a VRRS group transitions from the active state.

The following RADIUS attributes are included in VRRS accounting messages by default:

- Attribute 4, NAS-IP-Address
- Attribute 26, Cisco VSA Type 1, VRRS Name
- Attribute 40, Acct-Status-Type
- Attribute 41, Acct-Delay-Time
- Attribute 44, Acct-Session-Id

Accounting messages for a VRRS transitioning out of active state are sent after all PPPoE accounting stop messages for sessions that are part of that VRRS.

AAA Accounting Enhancements

AAA Broadcast Accounting

AAA broadcast accounting allows accounting information to be sent to multiple AAA servers at the same time; that is, accounting information can be broadcast to one or more AAA servers simultaneously. This functionality allows service providers to send accounting information to their own private AAA servers and to the AAA servers of their end customers. It also provides redundant billing information for voice applications.

Broadcasting is allowed among groups of RADIUS or TACACS+ servers, and each server group can define its backup servers for failover independently of other groups.

Thus, service providers and their end customers can use different protocols (RADIUS or TACACS+) for the accounting server. Service providers and their end customers can also specify their backup servers independently. As for voice applications, redundant accounting information can be managed independently through a separate group with its own failover sequence.

AAA Session MIB

The AAA session MIB feature allows customers to monitor and terminate their authenticated client connections using Simple Network Management Protocol (SNMP). The data of the client is presented so that it correlates directly to the AAA Accounting information reported by either the RADIUS or the TACACS+ server. AAA session MIB provides the following information:

- Statistics for each AAA function (when used in conjunction with the **show radius statistics** command)
- Status of servers providing AAA functions
- Identities of external AAA servers
- Real-time information (such as idle times), providing additional criteria for use by SNMP networks for assessing whether or not to terminate an active call



Note This command is supported only on Cisco AS5300 and Cisco AS5800 universal access server platforms.

The table below shows the SNMP user-end data objects that can be used to monitor and terminate authenticated client connections with the AAA session MIB feature.

Table 121: SNMP End-User Data Objects

SessionId	The session identification used by the AAA Accounting protocol (same value as reported by RADIUS attribute 44 (Acct-Session-ID)).
UserId	The user login ID or zero-length string if a login is unavailable.
IpAddr	The IP address of the session or 0.0.0.0 if an IP address is not applicable or unavailable.
IdleTime	The elapsed time in seconds that the session has been idle.

Disconnect	The session termination object used to disconnect the given client.
CallId	The entry index corresponding to this accounting session that the Call Tracker record stored.

The table below describes the AAA summary information provided by the AAA session MIB feature using SNMP on a per-system basis.

Table 122: SNMP AAA Session Summary

ActiveTableEntries	Number of sessions currently active.
ActiveTableHighWaterMark	Maximum number of sessions present at once since last system reinstallation.
TotalSessions	Total number of sessions since last system reinstallation.
DisconnectedSessions	Total number of sessions that have been disconnected using since last system reinstallation.

Accounting Attribute-Value Pairs

The network access server monitors the accounting functions defined in either TACACS+ AV pairs or RADIUS attributes, depending on which security method is implemented.

How to Configure Accounting

Configuring AAA Accounting Using Named Method Lists

To configure AAA Accounting using named method lists, perform the following steps:



Note System accounting does not use named method lists. For system accounting, define only the default method list.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **aaa accounting** {system | network | exec | connection | commands *level*} {default | *list-name*} {start-stop | stop-only | none} [*method1* [*method2...*]]
4. Do one of the following:
 - **line** [aux | console | tty | vty] *line-number* [*ending-line-number*]
 - **interface** *interface-type interface-number*
5. Do one of the following:
 - **accounting** {arap | commands *level* | connection | exec} {default | *list-name*}
 - **ppp accounting** {default | *list-name*}

6. Device(config-line)# end

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	aaa accounting {system network exec connection commands level} {default list-name} {start-stop stop-only none} [method1 [method2...]] Example: Device(config)# aaa accounting system default start-stop	Creates an accounting method list and enables accounting. The argument <i>list-name</i> is a character string used to name the created list.
Step 4	Do one of the following: <ul style="list-style-type: none"> • line [aux console tty vty] <i>line-number</i> [<i>ending-line-number</i>] • interface <i>interface-type interface-number</i> Example: Device(config)# line aux line1	Enters the line configuration mode for the lines to which the accounting method list is applied. or Enters the interface configuration mode for the interfaces to which the accounting method list is applied.
Step 5	Do one of the following: <ul style="list-style-type: none"> • accounting {arap commands level connection exec} {default list-name} • ppp accounting{default list-name} Example: Device(config-line)# accounting arap default	Applies the accounting method list to a line or set of lines. or Applies the accounting method list to an interface or set of interfaces.
Step 6	Device(config-line)# end Example: Device(config-line)# end	(Optional) Exits line configuration mode and returns to global configuration mode.

Configuring RADIUS System Accounting

Perform this task to configure RADIUS system accounting on the global RADIUS server:

SUMMARY STEPS

1. enable
2. configure terminal
3. aaa new-model
4. radius-server accounting system host-config
5. aaa group server radius *server-name*
6. server-private {*host-name* | *ip-address*} key {[0 *server-key* | 7 *server-key*] *server-key*}
7. accounting system host-config
8. end

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	aaa new-model Example: Device(config)# aaa new-model	Enables AAA network security services.
Step 4	radius-server accounting system host-config Example: Device(config)# radius-server accounting system host-config	Enables the device to send a system accounting record for the addition and deletion of a RADIUS server.
Step 5	aaa group server radius <i>server-name</i> Example: Device(config)# aaa group server radius radgroup1	Adds the RADIUS server and enters server-group configuration mode. <ul style="list-style-type: none"> • The <i>server-name</i> argument specifies the RADIUS server group name.
Step 6	server-private {<i>host-name</i> <i>ip-address</i>} key {[0 <i>server-key</i> 7 <i>server-key</i>] <i>server-key</i>} Example: Device(config-sg-radius)# server-private 172.16.1.11 key cisco	Enters the hostname or IP address of the RADIUS server and hidden server key. <ul style="list-style-type: none"> • (Optional) 0 with the <i>server-key</i> argument specifies that an unencrypted (cleartext) hidden server key follows. • (Optional) 7 with the <i>server-key</i> argument specifies that an encrypted hidden server key follows.

	Command or Action	Purpose
		<ul style="list-style-type: none"> The <i>server-key</i> argument specifies the hidden server key. If the <i>server-key</i> argument is configured without the 0 or 7 preceding it, it is unencrypted. <p>Note Once the server-private command is configured, RADIUS system accounting is enabled.</p>
Step 7	accounting system host-config Example: <pre>Device(config-sg-radius)# accounting system host-config</pre>	Enables the generation of system accounting records for private server hosts when they are added or deleted.
Step 8	end Example: <pre>Device(config-sg-radius)# end</pre>	Exits server-group configuration mode and returns to privileged EXEC mode.

Suppressing Generation of Accounting Records for Null Username Sessions

When AAA Accounting is activated, the Cisco IOS software issues accounting records for all users on the system, including users whose username string, because of protocol translation, is NULL. An example of this is users who come in on lines where the **aaa authentication login *method-list* none** command is applied. To prevent accounting records from being generated for sessions that do not have usernames associated with them, use the following command in global configuration mode:

Command	Purpose
<pre>Device(config)# aaa accounting suppress null-username</pre>	Prevents accounting records from being generated for users whose username string is NULL.

Generating Interim Accounting Records

To enable periodic interim accounting records to be sent to the accounting server, use the following command in global configuration mode:

Command	Purpose
<pre>Device(config)# aaa accounting update [newinfo] [periodic] number</pre>	Enables periodic interim accounting records to be sent to the accounting server.

When the **aaa accounting update** command is activated, the Cisco IOS software issues interim accounting records for all users on the system. If the keyword **newinfo** is used, interim accounting records are sent to the accounting server every time there is new accounting information to report. An example of this would be when IPCP completes IP address negotiation with the remote peer. The interim accounting record includes the negotiated IP address used by the remote peer.

When used with the keyword **periodic**, interim accounting records are sent periodically as defined by the *number* argument. The interim accounting record contains all of the accounting information recorded for that user up to the time the interim accounting record is sent.



Caution Using the **aaa accounting update periodic** command can cause heavy congestion when many users are logged in to the network.

Generating Accounting Records for Failed Login or Session

When AAA Accounting is activated, the Cisco IOS software does not generate accounting records for system users who fail login authentication, or who succeed in login authentication but fail PPP negotiation for some reason.

To specify that accounting stop records be generated for users who fail to authenticate at login or during session negotiation, use the following command in global configuration mode:

Command	Purpose
Device(config)# aaa accounting send stop-record authentication failure	Generates “stop” records for users who fail to authenticate at login or during session negotiation using PPP.
Device(config)# aaa accounting send stop-record always	Sends authentication, authorization, and accounting (AAA) stop records regardless of whether a start record was sent earlier.

Specifying Accounting NETWORK-Stop Records Before EXEC-Stop Records

For PPP users who start EXEC terminal sessions, you can specify the NETWORK records to be generated before EXEC-stop records. In cases such as billing customers for specific services, it can be desirable to keep network start and stop records together, essentially “nesting” them within the framework of the EXEC start and stop messages. For example, a user dialing in using PPP can create the following records: EXEC-start, NETWORK-start, EXEC-stop, NETWORK-stop. By nesting the accounting records, NETWORK-stop records follow NETWORK-start messages: EXEC-start, NETWORK-start, NETWORK-stop, EXEC-stop.

To nest accounting records for user sessions, use the following command in global configuration mode:

Command	Purpose
Device(config)# aaa accounting nested	Nests network accounting records.

Configuring AAA Resource Failure Stop Accounting

To enable resource failure stop accounting, use the following command in global configuration mode:

Command	Purpose
<pre>Device(config)# aaa accounting resource <i>method-list</i> stop-failure group <i>server-group</i></pre>	<p>Generates a “stop” record for any calls that do not reach user authentication.</p> <p>Note Before configuring this feature, the tasks described in the Prerequisites for Configuring Accounting, on page 1149 section must be performed, and SNMP must be enabled on the network access server.</p>

Configuring AAA Resource Accounting for Start-Stop Records

To enable full resource accounting for start-stop records, use the following command in global configuration mode:

Command	Purpose
<pre>Device(config)# aaa accounting resource <i>method-list</i> start-stop group <i>server-group</i></pre>	<p>Supports the ability to send a “start” record at each call setup, followed with a corresponding “stop” record at the call disconnect.</p> <p>Note Before configuring this feature, the tasks described in the Prerequisites for Configuring Accounting, on page 1149 section must be performed, and SNMP must be enabled on the network access server.</p>

Configuring AAA Broadcast Accounting

To configure AAA broadcast accounting, use the **aaa accounting** command in global configuration mode:

Command	Purpose
<pre>Device(config)# aaa accounting {system network exec connection commands <i>level</i>} {default <i>list-name</i>} {start-stop stop-only none} [broadcast] <i>method1</i> [<i>method2...</i>]</pre>	<p>Enables sending accounting records to multiple AAA servers. Simultaneously sends accounting records to the first server in each group. If the first server is unavailable, failover occurs using the backup servers defined within that group.</p>

Configuring Per-DNIS AAA Broadcast Accounting

To configure AAA broadcast accounting per DNIS, use the **aaa dnis map accounting network** command in global configuration mode:

Command	Purpose
<pre>Device(config)# aaa dnis map dnis-number accounting network [start-stop stop-only none] [broadcast] <i>method1</i> [<i>method2...</i>]</pre>	<p>Allows per-DNIS accounting configuration. This command has precedence over the global aaa accounting command.</p> <p>Enables sending accounting records to multiple AAA servers. Simultaneously sends accounting records to the first server in each group. If the first server is unavailable, failover occurs using the backup servers defined within that group.</p>

Configuring AAA Session MIB

The following tasks must be performed before configuring the AAA session MIB feature:

- Configure SNMP.
- Configure AAA.
- Define the RADIUS or TACACS+ server characteristics.



Note Overusing SNMP can affect the overall system performance; therefore, normal network management performance must be considered when this feature is used.

To configure AAA session MIB, use the following command in global configuration mode

SUMMARY STEPS

1. Device (config)# **aaa session-mib disconnect**

DETAILED STEPS

	Command or Action	Purpose
Step 1	Device (config)# aaa session-mib disconnect	Monitors and terminates authenticated client connections using SNMP. To terminate the call, the disconnect keyword must be used.

Configuring VRRS Accounting

Perform the following task to configure Virtual Router Redundancy Service (VRRS) to send AAA Accounting messages to the AAA server:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **aaa accounting vrrs** {**default** | *list-name*} **start-stop** *method1* [*method2...*]
4. **aaa attribute list** *list-name*

5. **attribute type** *name value* [**service** *service*] [**protocol** *protocol*][**mandatory**][**tag** *tag-value*]
6. **exit**
7. **vrrs** *vrrs-group-name*
8. **accounting delay** *seconds*
9. **accounting method** {**default** | *accounting-method-list*}
10. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	aaa accounting vrrs { default <i>list-name</i> } start-stop <i>method1</i> [<i>method2...</i>] Example: Device(config)# aaa accounting vrrs default start-stop	Enables AAA accounting for VRRS.
Step 4	aaa attribute list <i>list-name</i> Example: Device(config)# aaa attribute list list1	Defines a AAA attribute list locally on a device, and enters attribute list configuration mode.
Step 5	attribute type <i>name value</i> [service <i>service</i>] [protocol <i>protocol</i>][mandatory][tag <i>tag-value</i>] Example: Device(config-attr-list)# attribute type example 1	Defines an attribute type that is to be added to an attribute list locally on a device.
Step 6	exit Example: Device(config-attr-list)# exit	Exits attribute list configuration mode and returns to global configuration mode.
Step 7	vrrs <i>vrrs-group-name</i> Example: Device(config)# vrrs vrrs1	(Optional) Defines a VRRP group and configures parameters for the VRRS group, and enters VRRS configuration mode.

	Command or Action	Purpose
Step 8	accounting delay <i>seconds</i> Example: Device(config-vrrs)# accounting delay 10	(Optional) Specifies the delay time for sending accounting-off messages to the VRRS.
Step 9	accounting method { default <i>accounting-method-list</i> } Example: Device(config-vrrs)# accounting method default	(Optional) Enables VRRS accounting for a VRRP group.
Step 10	end Example: Device(config-vrrs)# end	Exits VRRS configuration mode and returns to privileged EXEC mode.

Establishing a Session with a Device if the AAA Server is Unreachable

To establish a console or telnet session with a device if the AAA server is unreachable, use the following command in global configuration mode:

Command	Purpose
Device(config)# no aaa accounting system guarantee-first	<p>The aaa accounting system guarantee-first command guarantees system accounting as the first record, which is the default condition.</p> <p>In some situations, users may be prevented from starting a session on the console or terminal connection until after the system reloads, which can take more than three minutes. To resolve this problem, the no aaa accounting system guarantee-first command can be used.</p>



Note Entering the **no aaa accounting system guarantee-first** command is not the only condition by which the console or telnet session can be started. For example, if the privileged EXEC session is being authenticated by TACACS and the TACACS server is not reachable, then the session cannot start.

Monitoring Accounting

No specific **show** command exists for either RADIUS or TACACS+ accounting. To obtain accounting records displaying information about users currently logged in, use the following command in privileged EXEC mode:

Command	Purpose
Device# show accounting	Allows display of the active accountable events on the network and helps collect information in the event of a data loss on the accounting server.

Troubleshooting Accounting

To troubleshoot accounting information, use the following command in privileged EXEC mode:

Command	Purpose
Device# debug aaa accounting	Displays information on accountable events as they occur.

Configuration Examples for Accounting

Example Configuring Named Method List

The following example shows how to configure a Cisco AS5200 (enabled for AAA and communication with a RADIUS security server) in order for AAA services to be provided by the RADIUS server. If the RADIUS server fails to respond, then the local database is queried for authentication and authorization information, and accounting services are handled by a TACACS+ server.

```

aaa new-model
aaa authentication login admins local
aaa authentication ppp dialins group radius local
aaa authorization network blue1 group radius local
aaa accounting network red1 start-stop group radius group tacacs+
username root password ALongPassword
tacacs-server host 172.31.255.0
tacacs-server key goaway
radius-server host 172.16.2.7
radius-server key myRaDiUSpassWoRd
interface group-async 1
  group-range 1 16
  encapsulation ppp
  ppp authentication chap dialins
  ppp authorization blue1
  ppp accounting red1
line 1 16
  autoselect ppp
  autoselect during-login
  login authentication admins
  modem dialin

```

The lines in this sample RADIUS AAA configuration are defined as follows:

- The **aaa new-model** command enables AAA network security services.
- The **aaa authentication login admins local** command defines a method list “admins”, for login authentication.
- The **aaa authentication ppp dialins group radius local** command defines the authentication method list “dialins”, which specifies that first RADIUS authentication and then (if the RADIUS server does not respond) local authentication is used on serial lines using PPP.
- The **aaa authorization network blue1 group radius local** command defines the network authorization method list named “blue1”, which specifies that RADIUS authorization is used on serial lines using PPP. If the RADIUS server fails to respond, then local network authorization is performed.

- The **aaa accounting network red1 start-stop group radius group tacacs+** command defines the network accounting method list named red1, which specifies that RADIUS accounting services (in this case, start and stop records for specific events) are used on serial lines using PPP. If the RADIUS server fails to respond, accounting services are handled by a TACACS+ server.
- The **username** command defines the username and password to be used for the PPP Password Authentication Protocol (PAP) caller identification.
- The **tacacs-server host** command defines the name of the TACACS+ server host.
- The **tacacs-server key** command defines the shared secret text string between the network access server and the TACACS+ server host.
- The **radius-server host** command defines the name of the RADIUS server host.
- The **radius-server key** command defines the shared secret text string between the network access server and the RADIUS server host.
- The **interface group-async** command selects and defines an asynchronous interface group.
- The **group-range** command defines the member asynchronous interfaces in the interface group.
- The **encapsulation ppp** command sets PPP as the encapsulation method used on the specified interfaces.
- The **ppp authentication chap dialins** command selects Challenge Handshake Authentication Protocol (CHAP) as the method of PPP authentication and applies the “dialins” method list to the specified interfaces.
- The **ppp authorization blue1** command applies the blue1 network authorization method list to the specified interfaces.
- The **ppp accounting red1** command applies the red1 network accounting method list to the specified interfaces.
- The **line** command switches the configuration mode from global configuration to line configuration and identifies the specific lines being configured.
- The **autoselect ppp** command configures the Cisco IOS software to allow a PPP session to start up automatically on these selected lines.
- The **autoselect during-login** command is used to display the username and password prompt without pressing the Return key. After the user logs in, the autoselect function (in this case, PPP) begins.
- The **login authentication admins** command applies the admins method list for login authentication.
- The **modem dialin** command configures modems attached to the selected lines to only accept incoming calls.

The **show accounting** command yields the following output for the preceding configuration:

```
Active Accounted actions on tty1, User username2 Priv 1
Task ID 5, Network Accounting record, 00:00:52 Elapsed
task_id=5 service=ppp protocol=ip address=10.0.0.98
```

The table below describes the fields contained in the preceding output.

Table 123: show accounting Field Descriptions

Field	Description
Active Accounted actions on	Terminal line or interface name user with which the user logged in.
User	User's ID.
Priv	User's privilege level.
Task ID	Unique identifier for each accounting session.
Accounting record	Type of accounting session.
Elapsed	Length of time (hh:mm:ss) for this session type.
attribute=value	AV pairs associated with this accounting session.

Example Configuring AAA Resource Accounting

The following example shows how to configure the resource failure stop accounting and resource accounting for start-stop records functions:

```
!Enable AAA on your network access server.
aaa new-model
!Enable authentication at login and list the AOL string name to use for login authentication.
aaa authentication login AOL group radius local
!Enable authentication for ppp and list the default method to use for PPP authentication.
aaa authentication ppp default group radius local
!Enable authorization for all exec sessions and list the AOL string name to use for
authorization.
aaa authorization exec AOL group radius if-authenticated
!Enable authorization for all network-related service requests and list the default method
to use for all network-related authorizations.
aaa authorization network default group radius if-authenticated
!Enable accounting for all exec sessions and list the default method to use for all start-stop
accounting services.
aaa accounting exec default start-stop group radius
!Enable accounting for all network-related service requests and list the default method to
use for all start-stop accounting services.
aaa accounting network default start-stop group radius
!Enable failure stop accounting.
aaa accounting resource default stop-failure group radius
!Enable resource accounting for start-stop records.
aaa accounting resource default start-stop group radius
```

Example Configuring AAA Broadcast Accounting

The following example shows how to turn on broadcast accounting using the global **aaa accounting** command:

```
aaa group server radius isp
server 10.0.0.1
server 10.0.0.2
aaa group server tacacs+ isp_customer
server 172.0.0.1
aaa accounting network default start-stop broadcast group isp group isp_customer
```



```
radius-server host 10.0.0.1
radius-server host 10.0.0.2
radius-server key key1
tacacs-server host 172.0.0.1 key key2
```

The **broadcast** keyword causes “start” and “stop” accounting records for network connections to be sent simultaneously to server 10.0.0.1 in the group `isp` and to server 172.0.0.1 in the group `isp_customer`. If server 10.0.0.1 is unavailable, failover to server 10.0.0.2 occurs. If server 172.0.0.1 is unavailable, no failover occurs because backup servers are not configured for the group `isp_customer`.

Example Configuring Per-DNIS AAA Broadcast Accounting

The following example shows how to turn on per DNIS broadcast accounting using the global `aaa dnis map accounting network` command:

```
aaa group server radius isp
  server 10.0.0.1
  server 10.0.0.2
aaa group server tacacs+ isp_customer
  server 172.0.0.1
aaa dnis map enable
aaa dnis map 7777 accounting network start-stop broadcast group isp group isp_customer
radius-server host 10.0.0.1
radius-server host 10.0.0.2
radius-server key key_1
tacacs-server host 172.0.0.1 key key_2
```

The **broadcast** keyword causes “start” and “stop” accounting records for network connection calls having DNIS number 7777 to be sent simultaneously to server 10.0.0.1 in the group `isp` and to server 172.0.0.1 in the group `isp_customer`. If server 10.0.0.1 is unavailable, failover to server 10.0.0.2 occurs. If server 172.0.0.1 is unavailable, no failover occurs because backup servers are not configured for the group `isp_customer`.

Example AAA Session MIB

The following example shows how to set up the AAA session MIB feature to disconnect authenticated client connections for PPP users:

```
aaa new-model
aaa authentication ppp default group radius
aaa authorization network default group radius
aaa accounting network default start-stop group radius
aaa session-mib disconnect
```

Example Configuring VRRS Accounting

The following example shows how to configure VRRS to send AAA Accounting messages to the AAA server:

```
Router# configure terminal
Router(config)# aaa accounting vrrs vrrp-mlist-1 start-stop group radius
Router(config)# aaa attribute list vrrp-1-attr
Router(config-attr-list)# attribute type account-delay 10
Router(config-attr-list)# exit
Router(config)# vrrs vrrp-group-1
Router(config-vrrs)# accounting delay 10
```

```
Router(config-vrrs) # accounting method vrrp-mlist-1
Router(config-vrrs) # exit
```

Additional References for Configuring Accounting

Related Documents

Related Topic	Document Title
Cisco security commands	<ul style="list-style-type: none"> • Cisco IOS Security Command Reference: Commands A to C • Cisco IOS Security Command Reference: Commands D to L • Cisco IOS Security Command Reference: Commands M to R • Cisco IOS Security Command Reference: Commands S to Z

RFCs

RFC	Title
<i>RFC 2903</i>	<i>Generic AAA Architecture</i>
<i>RFC 2904</i>	<i>AAA Authorization Framework</i>
<i>RFC 2906</i>	<i>AAA Authorization Requirements</i>
<i>RFC 2989</i>	<i>Criteria for Evaluating AAA Protocols for Network Access</i>

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

Feature Information for Configuring Accounting

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Table 124: Feature Information for Configuring Accounting

Feature Name	Releases	Feature Information
AAA Broadcast Accounting	Cisco IOS 15.2(1)E	AAA broadcast accounting allows accounting information to be sent to multiple AAA servers at the same time; that is, accounting information can be broadcast to one or more AAA servers simultaneously.
AAA Resource Accounting for Start-Stop Records	Cisco IOS 15.2(1)E	AAA resource accounting for start-stop records supports the ability to send a “start” record at each call setup, followed by a corresponding “stop” record at the call disconnect. This functionality can be used to manage and monitor wholesale customers from one source of data reporting, such as accounting records.
AAA Session MIB	Cisco IOS 15.2(1)E	The AAA session MIB feature allows customers to monitor and terminate their authenticated client connections using SNMP. The data of the client is presented so that it correlates directly to the AAA Accounting information reported by either the RADIUS or the TACACS+ server.
AAA: IPv6 Accounting Delay Enhancements	Cisco IOS 15.2(1)E	VRRS provides a multi-client information abstraction and management service between a First Hop Redundancy Protocol (FHRP) and a registered client.



CHAPTER 57

Configuring Local Authentication and Authorization

- [How to Configure Local Authentication and Authorization, on page 1179](#)
- [Monitoring Local Authentication and Authorization, on page 1181](#)
- [Additional References, on page 1182](#)
- [Feature Information for Local Authentication and Authorization, on page 1182](#)

How to Configure Local Authentication and Authorization

Configuring the Switch for Local Authentication and Authorization

You can configure AAA to operate without a server by setting the switch to implement AAA in local mode. The switch then handles authentication and authorization. No accounting is available in this configuration.



Note To secure the switch for HTTP access by using AAA methods, you must configure the switch with the **ip http authentication aaa** global configuration command. Configuring AAA authentication does not secure the switch for HTTP access by using AAA methods.

Follow these steps to configure AAA to operate without a server by setting the switch to implement AAA in local mode:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **aaa new-model**
4. **aaa authentication login default local**
5. **aaa authorization exec default local**
6. **aaa authorization network default local**
7. **username** *name* [**privilege level**] {**password** *encryption-type password*}
8. **end**
9. **show running-config**

10. copy running-config startup-config

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	aaa new-model Example: Device(config)# aaa new-model	Enables AAA.
Step 4	aaa authentication login default local Example: Device(config)# aaa authentication login default local	Sets the login authentication to use the local username database. The default keyword applies the local user database authentication to all ports.
Step 5	aaa authorization exec default local Example: Device(config)# aaa authorization exec default local	Configures user AAA authorization, check the local database, and allow the user to run an EXEC shell.
Step 6	aaa authorization network default local Example: Device(config)# aaa authorization network default local	Configures user AAA authorization for all network-related service requests.
Step 7	username name [privilege level] {password encryption-type password} Example: Device(config)# username your_user_name privilege 1 password 7 secret567	Enters the local database, and establishes a username-based authentication system. Repeat this command for each user. <ul style="list-style-type: none"> • For <i>name</i>, specify the user ID as one word. Spaces and quotation marks are not allowed.

	Command or Action	Purpose
		<ul style="list-style-type: none"> • (Optional) For <i>level</i>, specify the privilege level the user has after gaining access. The range is 0 to 15. Level 15 gives privileged EXEC mode access. Level 0 gives user EXEC mode access. • For <i>encryption-type</i>, enter 0 to specify that an unencrypted password follows. Enter 7 to specify that a hidden password follows. • For <i>password</i>, specify the password the user must enter to gain access to the switch. The password must be from 1 to 25 characters, can contain embedded spaces, and must be the last option specified in the username command.
Step 8	end Example: Device(config)# end	Returns to privileged EXEC mode.
Step 9	show running-config Example: Device# show running-config	Verifies your entries.
Step 10	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Monitoring Local Authentication and Authorization

To display Local Authentication and Authorization configuration, use the **show running-config** privileged EXEC command.

Additional References

Error Message Decoder

Description	Link
To help you research and resolve system error messages in this release, use the Error Message Decoder tool.	https://www.cisco.com/cgi-bin/Support/Errordecoder/index.cgi

MIBs

MIB	MIBs Link
All the supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
<p>The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies.</p> <p>To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds.</p> <p>Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.</p>	http://www.cisco.com/support

Feature Information for Local Authentication and Authorization

Release	Feature Information
Cisco IOS Release 15.0(2)EX1	This feature was introduced.



CHAPTER 58

Certification Authority Interoperability

This chapter describes how to configure certification authority (CA) interoperability, which is provided in support of the IPsec protocol. CA interoperability permits Cisco IOS devices and CAs to communicate so that your Cisco IOS device can obtain and use digital certificates from the CA. Although IPsec can be implemented in your network without the use of a CA, using a CA provides manageability and scalability for IPsec.

- [Prerequisites For Certification Authority, on page 1183](#)
- [Restrictions for Certification Authority, on page 1183](#)
- [Information About Certification Authority, on page 1183](#)
- [How to Configure Certification Authority, on page 1186](#)
- [Monitoring and Maintaining Certification Authority, on page 1194](#)

Prerequisites For Certification Authority

You need to have a certification authority (CA) available to your network before you configure this interoperability feature. The CA must support the Public Key Infrastructure (PKI) protocol, and the Simple Certificate Enrollment Protocol (SCEP).

Restrictions for Certification Authority

When configuring your CA, the following restrictions apply:

- This feature should be configured only when you also configure both IPsec and Internet Key Exchange (IKE) in your network.
- The Cisco IOS software does not support CA server public keys greater than 2048 bits.

Information About Certification Authority

CA Supported Standards

Without certification authority (CA) interoperability, Cisco IOS devices could not use CAs when deploying IPsec. CAs provide a manageable, scalable solution for IPsec networks.

Cisco supports the following standards with this feature:

- **IPSec**—IPSec is a framework of open standards that provides data confidentiality, data integrity, and data authentication between participating peers. IPSec provides these security services at the IP layer; it uses Internet Key Exchange to handle negotiation of protocols and algorithms based on local policy, and to generate the encryption and authentication keys to be used by IPSec. IPSec can be used to protect one or more data flows between a pair of hosts, between a pair of security gateways, or between a security gateway and a host.
- **Internet Key Exchange (IKE)**—A hybrid protocol that implements Oakley and Skeme key exchanges inside the Internet Security Association Key Management Protocol (ISAKMP) framework. Although IKE can be used with other protocols, its initial implementation is with the IPSec protocol. IKE provides authentication of the IPSec peers, negotiates IPSec keys, and negotiates IPSec security associations.
- **Public-Key Cryptography Standard #7 (PKCS #7)**—A standard from RSA Data Security, Inc., used to encrypt and sign certificate enrollment messages.
- **Public-Key Cryptography Standard #10 (PKCS #10)**—A standard syntax from RSA Data Security, Inc. for certificate requests.
- **RSA Keys**—RSA is the public key cryptographic system developed by Ron Rivest, Adi Shamir, and Leonard Adleman. RSA keys come in pairs: one public key and one private key.
- **X.509v3 certificates**—Certificate support that allows the IPSec-protected network to scale by providing the equivalent of a digital ID card to each device. When two devices wish to communicate, they exchange digital certificates to prove their identity (thus removing the need to manually exchange public keys with each peer or to manually specify a shared key at each peer). These certificates are obtained from a CA. X.509 is part of the X.500 standard of the ITU.

Purpose of CAs

Certificate authorities (CAs) are responsible for managing certificate requests and issuing certificates to participating IPSec network devices. These services provide centralized key management for the participating devices.

CAs simplify the administration of IPSec network devices. You can use a CA with a network containing multiple IPSec-compliant devices such as routers.

Digital signatures, enabled by public key cryptography, provide a means of digitally authenticating devices and individual users. In public key cryptography, such as the RSA encryption system, each user has a key pair containing both a public and a private key. The keys act as complements, and anything encrypted with one of the keys can be decrypted with the other. In simple terms, a signature is formed when data is encrypted with a user's private key. The receiver verifies the signature by decrypting the message with the sender's public key. The fact that the message could be decrypted using the sender's public key indicates that the holder of the private key, the sender, must have created the message. This process relies on the receiver's having a copy of the sender's public key and knowing with a high degree of certainty that it really does belong to the sender and not to someone pretending to be the sender.

Digital certificates provide the link. A digital certificate contains information to identify a user or device, such as the name, serial number, company, department, or IP address. It also contains a copy of the entity's public key. The certificate is itself signed by a certification authority (CA), a third party that is explicitly trusted by the receiver to validate identities and to create digital certificates.

In order to validate the signature of the CA, the receiver must first know the CA's public key. Normally this process is handled out-of-band or through an operation done at installation. For instance, most web browsers are configured with the public keys of several CAs by default. The Internet Key Exchange (IKE), an essential component of IPSec, can use digital signatures to scalably authenticate peer devices before setting up security associations.

Without digital signatures, one must manually exchange either public keys or secrets between each pair of devices that use IPsec to protect communications between them. Without certificates, every new device added to the network requires a configuration change on every other device with which it communicates securely. With digital certificates, each device is enrolled with a certification authority. When two devices wish to communicate, they exchange certificates and digitally sign data to authenticate each other. When a new device is added to the network, one simply enrolls that device with a CA, and none of the other devices needs modification. When the new device attempts an IPsec connection, certificates are automatically exchanged and the device can be authenticated.

Implementing IPsec Without CAs

Without a CA, if you want to enable IPsec services (such as encryption) between two Cisco devices, you must first ensure that each device has the key of the other device (such as an RSA public key or a shared key). This requirement means that you must manually perform one of the following operations:

- At each device, enter the RSA public key of the other device.
- At each device, specify a shared key to be used by both device.

In the above illustration, each device uses the key of the other device to authenticate the identity of the other device; this authentication always occurs when IPsec traffic is exchanged between the two devices.

If you have multiple Cisco devices in a mesh topology and wish to exchange IPsec traffic passing among all of those devices, you must first configure shared keys or RSA public keys among all of those devices.

Every time a new device is added to the IPsec network, you must configure keys between the new device and each of the existing devices. (In Figure 34, four additional two-part key configurations would be required to add a single encrypting device to the network.)

Consequently, the more devices there are that require IPsec services, the more involved the key administration becomes. This approach does not scale well for larger, more complex encrypting networks.

Implementing IPsec With CAs

With a CA, you do not have to configure keys between all the encrypting devices. Instead, you individually enroll each participating device with the CA, requesting a certificate for the device. When this has been accomplished, each participating device can dynamically authenticate all the other participating devices. This process is illustrated in the illustration.

To add a new IPsec device to the network, you need only configure that new device to request a certificate from the CA, instead of making multiple key configurations with all the other existing IPsec devices.

Implementing IPsec with Multiple Root CAs

With multiple root CAs, you no longer have to enroll a device with the CA that issued a certificate to a peer. Instead, you configure a device with multiple CAs that it trusts. Thus, a device can use a configured CA (a trusted root) to verify certificates offered by a peer that were not issued by the same CA defined in the identity of the device.

Configuring multiple CAs allows two or more devices enrolled under different domains (different CAs) to verify the identity of each other when using IKE to set up IPsec tunnels.

Through Simple Certificate Enrollment Protocol (SCEP), each device is configured with a CA (the enrollment CA). The CA issues a certificate to the device that is signed with the private key of the CA. To verify the

certificates of peers in the same domain, the device is also configured with the root certificate of the enrollment CA.

To verify the certificate of a peer from a different domain, the root certificate of the enrollment CA in the domain of the peer must be configured securely in the device.

During Internet Key Exchange (IKE) phase one signature verification, the initiator will send the responder a list of its CA certificates. The responder should send the certificate issued by one of the CAs in the list. If the certificate is verified, the device saves the public key contained in the certificate on its public key ring.

With multiple root CAs, VPN users can establish trust in one domain and easily and securely distribute it to other domains. Thus, the required private communication channel between entities authenticated under different domains can occur.

How CA Certificates Are Used by IPsec Devices

When two IPsec devices want to exchange IPsec-protected traffic passing between them, they must first authenticate each other—otherwise, IPsec protection cannot occur. The authentication is done with IKE.

Without a CA, a device authenticates itself to the remote device using either RSA-encrypted nonces or preshared keys. Both methods require that keys must have been previously configured between the two devices.

With a CA, a device authenticates itself to the remote device by sending a certificate to the remote device and performing some public key cryptography. Each device must send its own unique certificate that was issued and validated by the CA. This process works because the certificate of each device encapsulates the public key of the device, each certificate is authenticated by the CA, and all participating devices recognize the CA as an authenticating authority. This scheme is called IKE with an RSA signature.

Your device can continue sending its own certificate for multiple IPsec sessions, and to multiple IPsec peers until the certificate expires. When its certificate expires, the device administrator must obtain a new one from the CA.

CAs can also revoke certificates for devices that will no longer participate in IPsec. Revoked certificates are not recognized as valid by other IPsec devices. Revoked certificates are listed in a certificate revocation list (CRL), which each peer may check before accepting a certificate from another peer.

Registration Authorities

Some CAs have a registration authority (RA) as part of their implementation. An RA is essentially a server that acts as a proxy for the CA so that CA functions can continue when the CA is offline.

Some of the configuration tasks described in this document differ slightly, depending on whether your CA supports an RA.

How to Configure Certification Authority

Managing NVRAM Memory Usage

Certificates and certificate revocation lists (CRLs) are used by your device when a CA is used. Normally certain certificates and all CRLs are stored locally in the NVRAM of the device, and each certificate and CRL uses a moderate amount of memory.

The following certificates are normally stored at your device:

- Certificate of your device
- Certificate of the CA
- Root certificates obtained from CA servers (all root certificates are saved in RAM after the device has been initialized)
- Two registration authority (RA) certificates (only if the CA supports an RA)

CRLs are normally stored at your device according to the following conditions:

- If your CA does not support an RA, only one CRL gets stored in the device.
- If your CA supports an RA, multiple CRLs can be stored in the device.

In some cases, storing these certificates and CRLs locally will not present any difficulty. In other cases, memory might become a problem—particularly if the CA supports an RA and a large number of CRLs have to be stored on the device. If the NVRAM is too small to store root certificates, only the fingerprint of the root certificate is saved.

To save NVRAM space, specify that certificates and CRLs should not be stored locally, but should be retrieved from the CA when needed. This alternative will save NVRAM space but could result in a slight performance impact. To specify that certificates and CRLs should not be stored locally on your device, but should be retrieved when required, enable query mode.

If you do not enable query mode now, you can do it later even if certificates and CRLs have already been stored on the device. In this case, when you enable query mode, the stored certificates and CRLs are deleted from the device after you save the configuration. (If you copy the configuration to a TFTP site prior to enabling query mode, you can save any stored certificates and CRLs at the TFTP site.)

Before disabling query mode, perform the **copy system:running-config nvram:startup-config** command to save all current certificates and CRLs to NVRAM. Otherwise they could be lost during a reboot.

To specify that certificates and CRLs should not be stored locally on your device, but should be retrieved when required, enable query mode by using the following command in global configuration mode:



Note Query mode may affect availability if the CA is down.

SUMMARY STEPS

1. **crypto ca certificate query**

DETAILED STEPS

	Command or Action	Purpose
Step 1	crypto ca certificate query Example: Device(config)# <code>crypto ca certificate query</code>	Enables query mode, which causes certificates and CRLs not to be stored locally.

Configuring the Device Host Name and IP Domain Name

You must configure the host name and IP domain name of a device if this has not already been done. This is required because the device assigns a fully qualified domain name (FQDN) to the keys and certificates used by IPsec, and the FQDN is based on the host name and IP domain name assigned to the device. For example, a certificate named "device20.example.com" is based on a device host name of "device20" and a device IP domain name of "example.com".

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **hostname *name***
4. **ip domain-name *name***
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	hostname <i>name</i> Example: Device(config)# hostname device1	Configures the host name of the device.
Step 4	ip domain-name <i>name</i> Example: Device(config)# ip domain-name domain.com	Configures the IP domain name of the device.
Step 5	end Example: Device(config)# end	Exits global configuration and returns to privileged EXEC mode.

Generating an RSA Key Pair

Rivest, Shamir, and Adelman (RSA) key pairs are used to sign and encrypt IKE key management messages and are required before obtaining a certificate for your device.

SUMMARY STEPS

1. **enable**

2. **configure terminal**
3. **crypto key generate rsa [usage-keys]**
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	crypto key generate rsa [usage-keys] Example: Device(config)# crypto key generate rsa usage-keys	Generates an RSA key pair. <ul style="list-style-type: none"> • Use the usage-keys keyword to specify special-usage keys instead of general-purpose keys.
Step 4	end Example: Device(config)# end	Exits global configuration and returns to privileged EXEC mode.

Declaring a Certification Authority

You should declare one certification authority (CA) to be used by the device.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **crypto ca trustpoint *name***
4. **enrollment url *url***
5. **enrollment command**
6. **exit**
7. **crypto pki trustpoint *name***
8. **cr1 query ldap://*url*[:*port*]**
9. **enrollment {mode ra | retry count *number* | retry period *minutes* | url *url*}**
10. **enrollment {mode ra | retry count *number* | retry period *minutes* | url *url*}**
11. **revocation-check *method1* [*method2 method3*]**
12. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none">• Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	crypto ca trustpoint <i>name</i> Example: Device(config)# crypto ca trustpoint ka	Declares the certification authority (CA) that your device should use and enters the CA profile enroll configuration mode.
Step 4	enrollment url <i>url</i> Example: Device(ca-profile-enroll)# enrollment url http://entrust:81	Specifies the URL of the CA server to which enrollment requests are sent.
Step 5	enrollment command Example: Device(ca-profile-enroll)# enrollment command	Specifies the HTTP command that is sent to the CA for enrollment.
Step 6	exit Example: Device(ca-profile-enroll)# exit	Exit CA profile enroll configuration mode and returns to global configuration mode.
Step 7	crypto pki trustpoint <i>name</i> Example: Device(config)# crypto pki trustpoint ka	Declares the trustpoint that your device should use and enters Ca-trustpoint configuration mode.
Step 8	crl query ldap://<i>url</i>[:<i>port</i>] Example: Device(ca-trustpoint)# crl query ldap://bar.cisco.com:3899	Queries the certificate revocation list (CRL) to ensure that the certificate of the peer is not revoked.
Step 9	enrollment {<i>mode ra</i> <i>retry count number</i> <i>retry period minutes</i> <i>url url</i>} Example: Device(ca-trustpoint)# enrollment retry period 2	Specifies the enrollment wait period between certificate request retries.
Step 10	enrollment {<i>mode ra</i> <i>retry count number</i> <i>retry period minutes</i> <i>url url</i>} Example: Device(ca-trustpoint)# enrollment retry count 8	Specifies the number of times a device will resend a certificate request when it does not receive a response from the previous request.

	Command or Action	Purpose
Step 11	revocation-check <i>method1</i> [<i>method2 method3</i>] Example: Device(ca-trustpoint)# revocation-check crl ocsp	Checks the revocation status of a certificate.
Step 12	end Example: Device(ca-trustpoint)# end	Exit CA trustpoint configuration mode and returns to privileged EXEC mode.

Configuring a Root CA (Trusted Root)

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **crypto ca trustpoint** *name*
4. **revocation-check** *method1* [*method2 method3*]
5. **root tftp** *server-hostname filename*
6. **enrollment http-proxy** *hostname port-number*
7. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	crypto ca trustpoint <i>name</i> Example: Device(config)# crypto ca trustpoint ka	Declares the trustpoint that your device should use and enters CA trustpoint configuration mode.
Step 4	revocation-check <i>method1</i> [<i>method2 method3</i>] Example: Device(ca-trustpoint)# revocation-check ocsp	Checks the revocation status of a certificate.
Step 5	root tftp <i>server-hostname filename</i> Example: Device(ca-trustpoint)# root tftp server1 file1	Obtains the certification authority (CA) certificate via TFTP.

	Command or Action	Purpose
Step 6	enrollment http-proxy <i>hostname port-number</i> Example: Device(ca-trustpoint)# enrollment http-proxy host2 8080	Accesses the certification authority (CA) by HTTP through the proxy server.
Step 7	end Example: Device(ca-trustpoint)# end	Exits CA trustpoint configuration mode and returns to privileged EXEC mode.

Authenticating the CA

The device must authenticate the certification authority (CA). It does this by obtaining the self-signed certificate of the CA, which contains the public key of the CA. Because the certificate of the CA is self-signed (the CA signs its own certificate) the public key of the CA should be manually authenticated by contacting the CA administrator to compare the fingerprint of the CA certificate when you perform this step.

Perform the following task to get the public key of the CA:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **crypto pki authenticatename**
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	crypto pki authenticatename Example: Device(config)# crypto pki authenticate myca	Authenticates the CA by getting the certificate of the CA.
Step 4	end Example: Device(config)# end	Exits global configuration mode and returns to privileged EXEC mode.

Requesting Signed Certificates

You must obtain a signed certificate from the certification authority (CA) for each of the RSA key pairs on your device. If you generated general-purpose RSA keys, your device has only one RSA key pair and needs only one certificate. If you previously generated special-usage RSA keys, your device has two RSA key pairs and needs two certificates.

Perform the following task to request signed certificates from the CA:



Note If your device reboots after you have issued the **crypto pki enroll** command, but before you have received the certificates, you must reissue the command and notify the CA administrator.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **crypto pki enroll** *number*
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	crypto pki enroll <i>number</i> Example: Device(config)# crypto pki enroll myca	Obtains certificates for your device from the CA.
Step 4	end Example: Device(config)# end	Exits global configuration mode and returns to privileged EXEC mode.

What to do next

Saving Your Configuration

Always remember to save your work when you make configuration changes.

Use the **copy system:running-config nvram:startup-config** command to save your configuration. This command includes saving RSA keys to private NVRAM. RSA keys are not saved with your configuration when you use a **copy system:running-config rpc:** or **copy system:running-config tftp:** command.

Monitoring and Maintaining Certification Authority

Requesting a Certificate Revocation List

You can request a certificate revocation list (CRL) only if the certification authority (CA) does not support a registration authority (RA). The following task applies only when the CA does not support an RA.

When a device receives a certificate from a peer, your device will download a CRL from the CA. The device then checks the CRL to make sure the certificate that the peer sent has not been revoked. (If the certificate appears on the CRL, the device will not accept the certificate and will not authenticate the peer.)

A CRL can be reused with subsequent certificates until the CRL expires if query mode is off. If the device receives a peer's certificate after the applicable CRL has expired, the device will download the new CRL.

If the device has a CRL that has not yet expired, but you suspect that the contents of the CRL are out of date, you can request that the latest CRL be downloaded immediately to replace the old CRL.

•

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **crypto pki crl request *name***
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	crypto pki crl request <i>name</i> Example: Device(config)# crypto pki crl request myca	Requests that a new certificate revocation list (CRL) be obtained immediately from the CA.
Step 4	end Example: Device(config)# end	Exits global configuration mode and returns to privileged EXEC mode.

Querying a Certification Revocation List

You can query a certificate revocation list (CRL) only when you configure your device with a trusted root. When your device receives a certificate from a peer from another domain (with a different CA), the CRL downloaded from the CA of the device will not include certificate information about the peer. Therefore, you should check the CRL published by the configured root with the LDAP URL to ensure that the certificate of the peer has not been revoked.

If you would like CRL of the root certificate to be queried when the device reboots, you must enter the **cr1 query** command.

Perform the following task to query the CRL published by the configured root with the LDAP URL:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **crypto pki trustpoint *name***
4. **cr1 query ldap *://url* : [*port*]**
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	crypto pki trustpoint <i>name</i> Example: Device(ca-trustpoint)# crypto pki trustpoint mytp	Declares the trustpoint that your device should use and enters CA trustpoint configuration mode.
Step 4	cr1 query ldap <i>://url</i> : [<i>port</i>] Example: Device(ca-trustpoint)# cr1 query ldap://url:[port]	Queries the CRL to ensure that the certificate of the peer has not been revoked.
Step 5	end Example: Device(ca-trustpoint)# end	Exits CA trustpoint configuration mode and returns to privileged EXEC mode.

Deleting RSA Keys from a Device

Under certain circumstances you may want to delete RSA keys from your device. For example, if you believe the RSA keys were compromised in some way and should no longer be used, you should delete the keys.

]

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **crypto key zeroize rsa** [*key-pair-label*]
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none">• Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	crypto key zeroize rsa [<i>key-pair-label</i>] Example: Device(config)# crypto key zeroize rsa	Deletes all Rivest, Shamir, and Adelman (RSA) keys from your device.
Step 4	end Example: Device(config)# end	Exits global configuration mode and returns to privileged EXEC mode.

What to do next

After you delete RSA keys from the device, you should also complete the following two additional tasks:

- Ask the CA administrator to revoke the device certificates at the CA; you must supply the challenge password that you created when you originally obtained the device certificates with the **crypto pki enroll** command.
- Manually remove the device certificates from the device configuration.

Deleting Public Keys for a Peer

Under certain circumstances you may want to delete RSA public keys of peer devices from your device configuration. For example, if you no longer trust the integrity of the public key of a peer, you should delete the key.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **crypto key pubkey-chain rsa**

4. **no named key** *key-name* [**encryption** | **signature**]
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	crypto key pubkey-chain rsa Example: Device(config)# crypto key pubkey-chain rsa	Enters public key chain configuration mode, so that you can manually specify other devices' RSA public keys.
Step 4	no named key <i>key-name</i> [encryption signature] Example: Device(config-pubkey-c)# no named-key otherpeer.example.com	Deletes the RSA public key of a remote peer and enters public key configuration mode.
Step 5	end Example: Device(config-pubkey)# end	Exits public key configuration mode and returns to privileged EXEC mode.

Deleting Certificates from the Configuration

If the need arises, you can delete certificates that are saved in your device. Your device saves its own certificates, the certificate of the CA, and any RA certificates.

To delete the CA's certificate, you must remove the entire CA identity, which also removes all certificates associated with the CA—your router's certificate, the CA certificate, and any RA certificates.

SUMMARY STEPS

1. **enable**
2. **show crypto pki certificates**
3. **configure terminal**
4. **crypto pki certificate chain** *name*
5. **no certificate** *certificate-serial-number*
6. **exit**
7. **no crypto pki import** *name* **certificate**
8. **exit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	show crypto pki certificates Example: Device# show crypto pki certificates	Displays information about your device certificate, the certification authority (CA) certificate, and any registration authority (RA) certificates.
Step 3	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 4	crypto pki certificate chain <i>name</i> Example: Device(config)# crypto pki certificate chain myca	Enters certificate chain configuration mode.
Step 5	no certificate <i>certificate-serial-number</i> Example: Device(config-cert-chain)# no certificate 0123456789ABCDEF0123456789ABCDEF	Deletes the certificate.
Step 6	exit Example: Device(config-cert-chain)# exit	Exits certificate chain configuration mode and returns to global configuration mode.
Step 7	no crypto pki import <i>name</i> certificate Example: Device(config)# no crypto pki import MS certificate	Deletes a certificate manually.
Step 8	exit Example: Device(config)# exit	Exits global configuration mode and returns to privileged EXEC mode.

Viewing Keys and Certificates

Perform the following task to view keys and certificates:

SUMMARY STEPS

1. enable
2. show crypto key mypubkey rsa [*keyname*]
3. show crypto key pubkey-chain rsa
4. show crypto key pubkey-chain rsa [*name key-name* | *address key-address*]

5. **show crypto pki certificates**
6. **show crypto pki trustpoints**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	show crypto key mypubkey rsa [keyname] Example: Device# show crypto key mypubkey rsa [keyname]	Displays the RSA public keys configured on a device.
Step 3	show crypto key pubkey-chain rsa Example: Device# show crypto key pubkey-chain rsa	Displays the RSA public keys of the peer that are stored on a device.
Step 4	show crypto key pubkey-chain rsa [name key-name address key-address] Example: Device# show crypto key pubkey-chain rsa address 209.165.202.129	Displays the address of a specific key.
Step 5	show crypto pki certificates Example: Device# show crypto pki certificates	Displays information about the device certificate, the certification authority (CA) certificate, and any registration authority (RA) certificates
Step 6	show crypto pki trustpoints Example: Device# show crypto pki certificates	Displays trustpoints that are configured on a device.



CHAPTER 59

MAC Authentication Bypass

The MAC Authentication Bypass feature is a MAC-address-based authentication mechanism that allows clients in a network to integrate with the Cisco Identity Based Networking Services (IBNS) and Network Admission Control (NAC) strategy using the client MAC address. The MAC Authentication Bypass feature is applicable to the following network environments:

- Network environments in which a supplicant code is not available for a given client platform.
- Network environments in which the end client configuration is not under administrative control, that is, the IEEE 802.1X requests are not supported on these networks.
- [Prerequisites for Configuring MAC Authentication Bypass, on page 1201](#)
- [Information About MAC Authentication Bypass, on page 1202](#)
- [How to Configure MAC Authentication Bypass, on page 1203](#)
- [Configuration Examples for MAC Authentication Bypass, on page 1208](#)
- [Additional References for MAC Authentication Bypass, on page 1209](#)
- [Feature Information for MAC Authentication Bypass, on page 1210](#)

Prerequisites for Configuring MAC Authentication Bypass

IEEE 802.1x—Port-Based Network Access Control

You should understand the concepts of port-based network access control and have an understanding of how to configure port-based network access control on your Cisco platform.

RADIUS and ACLs

You should understand the concepts of the RADIUS protocol and have an understanding of how to create and apply access control lists (ACLs). For more information, see the documentation for your Cisco platform and the *Securing User Services Configuration Guide Library*.

The device must have a RADIUS configuration and be connected to the Cisco secure access control server (ACS). For more information, see the *User Guide for Secure ACS Appliance 3.2*.

Information About MAC Authentication Bypass

Overview of the Cisco IOS Auth Manager

The capabilities of devices connecting to a given network can be different, thus requiring that the network support different authentication methods and authorization policies. The Cisco IOS Auth Manager handles network authentication requests and enforces authorization policies regardless of authentication method. The Auth Manager maintains operational data for all port-based network connection attempts, authentications, authorizations, and disconnections and, as such, serves as a session manager.

The possible states for Auth Manager sessions are as follows:

- **Idle**—In the idle state, the authentication session has been initialized, but no methods have yet been run. This is an intermediate state.
- **Running**—A method is currently running. This is an intermediate state.
- **Authc Success**—The authentication method has run successfully. This is an intermediate state.
- **Authc Failed**—The authentication method has failed. This is an intermediate state.
- **Authz Success**—All features have been successfully applied for this session. This is a terminal state.
- **Authz Failed**—At least one feature has failed to be applied for this session. This is a terminal state.
- **No methods**—There were no results for this session. This is a terminal state.

Overview of the Configurable MAB Username and Password

A MAC Authentication Bypass (MAB) operation involves authentication using RADIUS Access-Request packets with both the username and password attributes. By default, the username and the password values are the same and contain the MAC address. The Configurable MAB Username and Password feature enables you to configure both the username and the password attributes in the following scenarios:

- To enable MAB for an existing large database that uses formatted username attributes, the username format in the client MAC needs to be configured. Use the **mab request format attribute 1** command to configure the username format.
- Some databases do not accept authentication if the username and password values are the same. In such instances, the password needs to be configured to ensure that the password is different from the username. Use the **mab request format attribute 2** command to configure the password.

The Configurable MAB Username and Password feature allows interoperability between the Cisco IOS Authentication Manager and the existing MAC databases and RADIUS servers. The password is a global password and hence is the same for all MAB authentications and interfaces. This password is also synchronized across all supervisor devices to achieve high availability.

If the password is not provided or configured, the password uses the same value as the username. The table below describes the formatting of the username and the password:

MAC Address	Username Format (Group Size, Separator)	Username	Password Configured	Password Created
08002b8619de	(1, :) (1, -) (1, .)	0:8:0:0:2:b:8:6:1:9:d:e 0-8-0-0-2-b-8-6-1-9-d-e 0.8.0.0.2.b.8.6.1.9.d.e	None	0:8:0:0:2:b:8:6:1:9:d:e 0-8-0-0-2-b-8-6-1-9-d-e 0.8.0.0.2.b.8.6.1.9.d.e
08002b8619de	(1, :) (1, -) (1, .)	0:8:0:0:2:b:8:6:1:9:d:e 0-8-0-0-2-b-8-6-1-9-d-e 0.8.0.0.2.b.8.6.1.9.d.e	Password	Password
08002b8619de	(2, :) (2, -) (2, .)	08:00:2b:86:19:de 08-00-2b-86-19-de 08.00.2b.86.19.de	None	08:00:2b:86:19:de 08-00-2b-86-19-de 08.00.2b.86.19.de
08002b8619de	(2, :) (2, -) (2, .)	08:00:2b:86:19:de 08-00-2b-86-19-de 08.00.2b.86.19.de	Password	Password
08002b8619de	(4, :) (4, -) (4, .)	0800:2b86:19de 0800-2b86-19de 0800.2b86.19de	None	0800:2b86:19de 0800-2b86-19de 0800.2b86.19de
08002b8619de	(4, :) (4, -) (4, .)	0800:2b86:19de 0800-2b86-19de 0800.2b86.19de	Password	Password
08002b8619de	(12, <not applicable>)	08002b8619de	None	08002b8619de
08002b8619de	(12, <not applicable>)	08002b8619de	Password	Password

How to Configure MAC Authentication Bypass

Enabling MAC Authentication Bypass

Perform this task to enable the MAC Authentication Bypass feature on an 802.1X port.

SUMMARY STEPS

1. enable
2. configure terminal

3. `interface type slot / port`
4. `mab`
5. `end`
6. `show authentication sessions interface type slot / port details`

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	interface type slot / port Example: Device(config)# interface GigabitEthernet 1/2/1	Enters interface configuration mode.
Step 4	mab Example: Device(config-if)# mab	Enables MAB.
Step 5	end Example: Device(config-if)# end	Returns to privileged EXEC mode.
Step 6	show authentication sessions interface type slot / port details Example: Device# show authentication session interface GigabitEthernet 1/2/1 details	Displays the interface configuration and the authenticator instances on the interface.

Enabling Reauthentication on a Port

By default, ports are not automatically reauthenticated. You can enable automatic reauthentication and specify how often reauthentication attempts are made.

SUMMARY STEPS

1. `enable`

2. **configure terminal**
3. **interface** *type slot / port*
4. **switchport**
5. **switchport mode access**
6. **authentication port-control auto**
7. **mab [eap]**
8. **authentication periodic**
9. **authentication timer reauthenticate** *{seconds | server}*
10. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	interface <i>type slot / port</i> Example: Device(config)# interface GigabitEthernet 1/2/1	Enters interface configuration mode.
Step 4	switchport Example: Device(config-if)# switchport	Places interface in Layer 2 switched mode.
Step 5	switchport mode access Example: Device(config-if)# switchport mode access	Sets the interface type as a nontrunking, nontagged single VLAN Layer 2 interface.
Step 6	authentication port-control auto Example: Device(config-if)# authentication port-control auto	Configures the authorization state of the port.
Step 7	mab [eap] Example: Device(config-if)# mab	Enables MAB.

	Command or Action	Purpose
Step 8	authentication periodic Example: Device(config-if)# authentication periodic	Enables reauthentication.
Step 9	authentication timer reauthenticate {seconds server} Example: Device(config-if)# authentication timer reauthenticate 900	Configures the time, in seconds, between reauthentication attempts.
Step 10	end Example: Device(config-if)# end	Exits interface configuration mode and returns to privileged EXEC mode.

Specifying the Security Violation Mode

When there is a security violation on a port, the port can be shut down or traffic can be restricted. By default, the port is shut down. You can configure the period of time for which the port is shut down.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *type slot / port*
4. **switchport**
5. **switchport mode access**
6. **authentication port-control auto**
7. **mab [eap]**
8. **authentication violation {restrict | shutdown}**
9. **authentication timer restart** *seconds*
10. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	configure terminal Example:	Enters global configuration mode.

	Command or Action	Purpose
	Device# configure terminal	
Step 3	interface <i>type slot / port</i> Example: Device(config)# interface GigabitEthernet 1/2/1	Enters interface configuration mode.
Step 4	switchport Example: Device(config-if)# switchport	Places interface in Layer 2 switched mode.
Step 5	switchport mode access Example: Device(config-if)# switchport mode access	Sets the interface type as a nontrunking, nontagged single VLAN Layer 2 interface.
Step 6	authentication port-control auto Example: Device(config-if)# authentication port-control auto	Configures the authorization state of the port.
Step 7	mab [eap] Example: Device(config-if)# mab	Enables MAB.
Step 8	authentication violation {restrict shutdown} Example: Device(config-if)# authentication violation shutdown	Configures the action to be taken when a security violation occurs on the port.
Step 9	authentication timer restart <i>seconds</i> Example: Device(config-if)# authentication timer restart 30	Configures the period of time, in seconds, after which an attempt is made to authenticate an unauthorized port.
Step 10	end Example: Device(config-if)# end	Exits interface configuration mode and returns to privileged EXEC mode.

Enabling Configurable MAB Username and Password

SUMMARY STEPS

1. `enable`
2. `configure terminal`
3. `mab request format attribute 1 groupsize {1 | 2 | 4 | 12} separator {- | : | .} [lowercase | uppercase]`
4. `mab request format attribute 2 [0 | 7] password`
5. `end`

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	mab request format attribute 1 groupsize {1 2 4 12} separator {- : .} [lowercase uppercase] Example: Device(config)# mab request format attribute 1 groupsize 2 separator :	Configures the username format for MAB requests.
Step 4	mab request format attribute 2 [0 7] password Example: Device(config)# mab request format attribute 2 password1	Configures a global password for all MAB requests.
Step 5	end Example: Device(config)# end	Returns to privileged EXEC mode.

Configuration Examples for MAC Authentication Bypass

Example: MAC Authentication Bypass Configuration

In the following example, the **mab** command has been configured to enable the MAC Authorization Bypass (MAB) feature on the specified interface. The optional **show authentication sessions** command has been enabled to display the interface configuration and the authentication instances on the interface.

```

Device> enable
Device# configure terminal
Device(config)# interface GigabitEthernet2/1
Device(config-if)# mab
Device(config-if)# end
Device# show authentication sessions interface GigabitEthernet2/1 details

```

Example: Enabling Configurable MAB Username and Password

The following example shows how to configure the username format and password for MAC Authentication Bypass (MAB). In this example, the username format is configured as a group of 12 hexadecimal digits with no separator and the global password as **password1**.

```

Device> enable
Device# configure terminal
Device(config)# mab request format attribute 1 groupsize 2 separator :
Device(config)# mab request format attribute 2 password1
Device(config)# end

```

Additional References for MAC Authentication Bypass

MIBs

MIB	MIBs Link
<ul style="list-style-type: none"> • CISCO-AUTH-FRAMEWORK-MIB • CISCO-MAC-AUTH-BYPASS-MIB • CISCO-PAE-MIB • IEEE8021-PAE-MIB 	<p>To locate and download MIBs for selected platforms, Cisco IOS software releases, and feature sets, use Cisco MIB Locator found at the following URL:</p> <p>http://www.cisco.com/go/mibs</p>

RFCs

RFC	Title
RFC 3580	<i>IEEE 802.1x Remote Authentication Dial In User Service (RADIUS)</i>

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

Feature Information for MAC Authentication Bypass

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Table 125: Feature Information for MAC Authentication Bypass

Feature Name	Releases	Feature Information
MAC Authentication Bypass (MAB)	Cisco IOS XE 3.2SE Cisco IOS XE 3.3SE Cisco IOS XE 3.5E Cisco IOS 15.2(1)E	The MAC Authentication Bypass feature is a MAC-address-based authentication mechanism that allows clients in a network to integrate with the Cisco IBNS and NAC strategy using the client MAC address. The following commands were introduced or modified: dot1x mac-auth-bypass , show dot1x interface .
Configurable MAB Username and Password	Cisco IOS 15.2(1)E	The Configurable MAB Username and Password feature enables you to configure MAC Authentication Bypass (MAB) username format and password to allow interoperability between the Cisco IOS Authentication Manager and existing MAC databases and RADIUS servers. The following commands were introduced or modified: mab request format attribute 1 , mab request format attribute 2 .



CHAPTER 60

Password Strength and Management for Common Criteria

The Password Strength and Management for Common Criteria feature is used to specify password policies and security mechanisms for storing, retrieving, and providing rules to specify user passwords.

For local users, the user profile and the password information with the key parameters are stored on the Cisco device, and this profile is used for local authentication of users. The user can be an administrator (terminal access) or a network user (for example, PPP users being authenticated for network access).

For remote users, where the user profile information is stored in a remote server, a third-party authentication, authorization, and accounting (AAA) server may be used for providing AAA services, both for administrative and network access.

- [Restrictions for Password Strength and Management for Common Criteria, on page 1211](#)
- [Information About Password Strength and Management for Common Criteria, on page 1211](#)
- [How to Configure Password Strength and Management for Common Criteria, on page 1213](#)
- [Configuration Examples for Password Strength and Management for Common Criteria, on page 1216](#)
- [Additional References for Password Strength and Management for Common Criteria, on page 1217](#)
- [Feature Information for Password Strength and Management for Common Criteria, on page 1217](#)

Restrictions for Password Strength and Management for Common Criteria

Only four concurrent users can log on to the system by using vty at any moment.

Information About Password Strength and Management for Common Criteria

Password Composition Policy

The password composition policy allows you to create passwords of any combination of upper and lowercase characters, numbers, and special characters that include “!”, “@”, “#”, “\$”, “%”, “^”, “&”, “*”, “(”, and “)”.

Password Length Policy

The administrator has the flexibility to set the password's minimum and maximum length. The recommended minimum password length is 8 characters. The administrator can specify both the minimum (1) and the maximum (64) length for the password.

Password Lifetime Policy

The security administrator can provide a configurable option for a password to have a maximum lifetime. If the lifetime parameter is not configured, the configured password will never expire. The maximum lifetime can be configured by providing the configurable value in years, months, days, hours, minutes, and seconds. The lifetime configuration will survive across reloads as it is a part of the configuration, but every time the system reboots, the password creation time will be updated to the new time. For example, if a password is configured with a lifetime of one month and on the 29th day, the system reboots, then the password will be valid for one month after the system reboots.

Password Expiry Policy

If the user attempts to log on and if the user's password credentials have expired, then the following happens:

1. The user is prompted to set the new password after successfully entering the expired password.
2. When the user enters the new password, the password is validated against the password security policy.
3. If the new password matches the password security policy, then the AAA database is updated, and the user is authenticated with the new password.
4. If the new password does not match the password security policy, then the user is prompted again for the password. From AAA perspective, there is no restriction on the number of retries. The number of retries for password prompt in case of unsuccessful authentication is controlled by the respective terminal access interactive module. For example, for telnet, after three unsuccessful attempts, the session will be terminated.

If the password's lifetime is not configured for a user and the user has already logged on and if the security administrator configures the lifetime for that user, then the lifetime will be set in the database. When the same user is authenticated the next time, the system will check for password expiry. The password expiry is checked only during the authentication phase.

If the user has been already authenticated and logged on to the system and if the password expires, then no action will be taken. The user will be prompted to change the password only during the next authentication for the same user.

Password Change Policy

The new password must contain a minimum of 4 character changes from the previous password. A password change can be triggered by the following scenarios:

- The security administrator wants to change the password.
- The user is trying to get authenticated using a profile, and the password for that profile has expired.

When the security administrator changes the password security policy and the existing profile does not meet the password security policy rules, no action will be taken if the user has already logged on to the system.

The user will be prompted to change the password only when the user tries to get authenticated using the profile that does not meet the password security restriction.

When the user changes the password, the lifetime parameters set by the security administrator for the old profile will be the lifetime parameters for the new password.

For noninteractive clients such as dot1x, when the password expires, appropriate error messages will be sent to the clients, and the clients must contact the security administrator to renew the password.

User Reauthentication Policy

Users are reauthenticated when they change their passwords.

When users change their passwords on expiry, they will be authenticated against the new password. In such cases, the actual authentication happens based on the previous credentials, and the new password is updated in the database.



Note Users can change their passwords only when they are logging on and after the expiry of the old password; however, a security administrator can change the user's password at any time.

Support for Framed (Noninteractive) Session

When a client such as dot1x uses the local database for authentication, the Password Strength and Management for Common Criteria feature will be applicable; however, upon password expiry, clients will not be able to change the password. An appropriate failure message will be sent to such clients, and the user must request the security administrator to change the password.

How to Configure Password Strength and Management for Common Criteria

Configuring the Password Security Policy

Perform this task to create a password security policy and to apply the policy to a specific user profile.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **aaa new-model**
4. **aaa common-criteria policy *policy-name***
5. **char-changes *number***
6. **max-length *number***
7. **min-length *number***
8. **numeric-count *number***
9. **special-case *number***

10. `exit`
11. `username username common-criteria-policy policy-name password password`
12. `end`

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	aaa new-model Example: Device(config)# aaa new-model	Enables AAA globally.
Step 4	aaa common-criteria policy <i>policy-name</i> Example: Device(config)# aaa common-criteria policy <i>policyl</i>	Creates the AAA security password policy and enters common criteria configuration policy mode.
Step 5	char-changes <i>number</i> Example: Device(config-cc-policy)# char-changes 4	(Optional) Specifies the number of changed characters between old and new passwords.
Step 6	max-length <i>number</i> Example: Device(config-cc-policy)# max-length 25	(Optional) Specifies the maximum length of the password.
Step 7	min-length <i>number</i> Example: Device(config-cc-policy)# min-length 8	(Optional) Specifies the minimum length of the password.
Step 8	numeric-count <i>number</i> Example: Device(config-cc-policy)# numeric-count 4	(Optional) Specifies the number of numeric characters in the password.

	Command or Action	Purpose
Step 9	special-case <i>number</i> Example: Device(config-cc-policy)# special-case 3	(Optional) Specifies the number of special characters in the password.
Step 10	exit Example: Device(config-cc-policy)# exit	(Optional) Exits common criteria configuration policy mode and returns to global configuration mode.
Step 11	username <i>username</i> common-criteria-policy <i>policy-name</i> password <i>password</i> Example: Device(config)# username user1 common-criteria-policy policy1 password password1	(Optional) Applies a specific policy and password to a user profile.
Step 12	end Example: Device(config)# end	Returns to privileged EXEC mode.

Verifying the Common Criteria Policy

Perform this task to verify all the common criteria security policies.

SUMMARY STEPS

1. **enable**
2. **show aaa common-criteria policy name** *policy-name*
3. **show aaa common-criteria policy all**

DETAILED STEPS

Step 1

enable

Enables privileged EXEC mode.

Example:

```
Device> enable
```

Step 2

show aaa common-criteria policy name *policy-name*

Displays the password security policy information for a specific policy.

Example:

```
Device# show aaa common-criteria policy name policy1

Policy name: policy1
Minimum length: 1
Maximum length: 64
Upper Count: 20
Lower Count: 20
Numeric Count: 5
Special Count: 2
Number of character changes 4
Valid forever. User tied to this policy will not expire.
```

Step 3 show aaa common-criteria policy all

Displays password security policy information for all the configured policies.

Example:

```
Device# show aaa common-criteria policy all
=====
Policy name: policy1
Minimum length: 1
Maximum length: 64
Upper Count: 20
Lower Count: 20
Numeric Count: 5
Special Count: 2
Number of character changes 4
Valid forever. User tied to this policy will not expire.
=====
Policy name: policy2
Minimum length: 1
Maximum length: 34
Upper Count: 10
Lower Count: 5
Numeric Count: 4
Special Count: 2
Number of character changes 2
Valid forever. User tied to this policy will not expire.
=====
```

Configuration Examples for Password Strength and Management for Common Criteria

Example: Password Strength and Management for Common Criteria

The following example shows how to create a common criteria security policy and apply the specific policy to a user profile:

```
Device> enable
Device# configure terminal
Device(config)# aaa new-model
Device(config)# aaa common-criteria policy policy1
Device(config-cc-policy)# char-changes 4
Device(config-cc-policy)# max-length 20
```

```

Device(config-cc-policy)# min-length 6
Device(config-cc-policy)# numeric-count 2
Device(config-cc-policy)# special-case 2
Device(config-cc-policy)# exit
Device(config)# username user1 common-criteria-policy policy1 password password1
Device(config)# end

```

Additional References for Password Strength and Management for Common Criteria

The following sections provide references related to the RADIUS Packet of Disconnect feature.

RFCs

RFC	Title
RFC 2865	<i>Remote Authentication Dial-in User Service</i>
RFC 3576	<i>Dynamic Authorization Extensions to RADIUS</i>

Technical Assistance

Description	Link
<p>The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies.</p> <p>To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds.</p> <p>Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.</p>	http://www.cisco.com/techsupport

Feature Information for Password Strength and Management for Common Criteria

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Table 126: Feature Information for Password Strength and Management for Common Criteria

Feature Name	Releases	Feature Information
Password Strength and Management for Common Criteria	Cisco IOS 15.0(2)SE Cisco IOS 15.2(1)E	<p>The Password Strength and Management for Common Criteria feature is used to specify password policies and security mechanisms for storing, retrieving, and providing rules to specify user passwords.</p> <p>The following commands were introduced or modified: aaa common-criteria policy, debug aaa common-criteria, and show aaa common-criteria policy.</p>



CHAPTER 61

AAA-SERVER-MIB Set Operation

The AAA-SERVER-MIB Set Operation feature allows the authentication, authorization, and accounting (AAA) server configuration to be extended or expanded by using the CISCO-AAA-SERVER-MIB to create and add new AAA servers, modify the “KEY” under the CISCO-AAA-SERVER-MIB, and delete the AAA server configuration.

- [Prerequisites for AAA-SERVER-MIB Set Operation, on page 1219](#)
- [Restrictions for AAA-SERVER-MIB Set Operation, on page 1219](#)
- [Information About AAA-SERVER-MIB Set Operation, on page 1219](#)
- [How to Configure AAA-SERVER-MIB Set Operation, on page 1220](#)
- [Configuration Examples for AAA-SERVER-MIB Set Operation, on page 1221](#)
- [Additional References for AAA-SERVER-MIB Set Operation, on page 1223](#)
- [Feature Information for AAA-SERVER-MIB Set Operation, on page 1223](#)

Prerequisites for AAA-SERVER-MIB Set Operation

AAA must have been enabled on the router, that is, the **aaa new-model** command must have been configured. If this configuration has not been accomplished, the set operation fails.

Restrictions for AAA-SERVER-MIB Set Operation

Currently, the CISCO SNMP set operation is supported only for the RADIUS protocol. Therefore, only RADIUS servers in global configuration mode can be added, modified, or deleted.

Information About AAA-SERVER-MIB Set Operation

CISCO-AAA-SERVER-MIB

The CISCO-AAA-SERVER-MIB provides that statistics reflect both the state of the AAA server operation with the server itself and of AAA communications with external servers. The CISCO-AAA-SERVER-MIB provides the following information:

- Statistics for each AAA operation

- Status of servers that are providing AAA functions
- Identities of external AAA servers

CISCO-AAA-SERVER-MIB Set Operation

With the SET operation, you can do the following:

- Create or add a new AAA server.
- Modify the KEY under the CISCO-AAA-SERVER-MIB. This “secret key” is used for secure connectivity to the AAA server, which is present with the network access server (NAS) and the AAA server.
- Delete the AAA server configuration.

How to Configure AAA-SERVER-MIB Set Operation

Configuring AAA-SERVER-MIB Set Operations

No special configuration is required for this feature. The Simple Network Management Protocol (SNMP) framework can be used to manage MIBs. See the Additional References section for a reference to configuring SNMP.

Verifying SNMP Values

SNMP values can be verified by performing the following steps.

SUMMARY STEPS

1. **enable**
2. **show running-config | include radius-server host**
3. **show aaa servers**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	show running-config include radius-server host Example: Device# show running-config include radius-server host	Displays all the RADIUS servers that are configured in the global configuration mode.

	Command or Action	Purpose
Step 3	show aaa servers Example: Device# show aaa servers	Displays information about the number of requests sent to and received from authentication, authorization, and accounting (AAA) servers.

Configuration Examples for AAA-SERVER-MIB Set Operation

RADIUS Server Configuration and Server Statistics Example

The following sample output shows the RADIUS server configuration and server statistics before and after the set operation.

Before the Set Operation

```
Device# show running-config | include radius-server host

! The following line is for server 1.
radius-server host 172.19.192.238 auth-port 2095 acct-port 2096 key cisco2
! The following line is for server 2.
radius-server host 172.19.192.238 auth-port 1645 acct-port 1646
```

Server Statistics

```
Device# show aaa servers

RADIUS: id 2, priority 1, host 172.19.192.238, auth-port 2095, acct-port 2096
State: current UP, duration 25s, previous duration 0s
  Dead: total time 0s, count 7
Authen: request 8, timeouts 8
  Response: unexpected 0, server error 0, incorrect 0, time 0ms
  Transaction: success 0, failure 2
Author: request 0, timeouts 0
  Response: unexpected 0, server error 0, incorrect 0, time 0ms
  Transaction: success 0, failure 0
Account: request 0, timeouts 0
  Response: unexpected 0, server error 0, incorrect 0, time 0ms
  Transaction: success 0, failure 0
Elapsed time since counters last cleared: 5m
RADIUS: id 3, priority 2, host 172.19.192.238, auth-port 1645, acct-port 1646
State: current UP, duration 5s, previous duration 0s
  Dead: total time 0s, count 2
Authen: request 8, timeouts 8
  Response: unexpected 0, server error 0, incorrect 0, time 0ms
  Transaction: success 0, failure 4
Author: request 0, timeouts 0
  Response: unexpected 0, server error 0, incorrect 0, time 0ms
  Transaction: success 0, failure 0
Account: request 0, timeouts 0
  Response: unexpected 0, server error 0, incorrect 0, time 0ms
  Transaction: success 0, failure 0
Elapsed time since counters last cleared: 3m
```

SNMP Get Operation to Check the Configuration and Statistics of the RADIUS Servers

```

aaa-server5:/users/smetri> getmany 10.0.1.42 casConfigTable
casAddress.2.2 = 172.19.192.238
casAddress.2.3 = 172.19.192.238
casAuthenPort.2.2 = 2095
casAuthenPort.2.3 = 1645
casAcctPort.2.2 = 2096
casAcctPort.2.3 = 1646
casKey.2.2 =
casKey.2.3 =
! The following line shows priority for server 1.
casPriority.2.2 = 1
! The following line shows priority for server 2.
casPriority.2.3 = 2
casConfigRowStatus.2.2 = active(1)
casConfigRowStatus.2.3 = active(1)
aaa-server5:/users/smetri>

```

SNMP Set Operation

The key of the existing RADIUS server is being changed. The index “1” is being used. That index acts as a wildcard for addition, deletion, or modification of any entries.

```

Change the key for server 1:=>
aaa-server5:/users/smetri> setany -v2c 10.0.1.42 public casAddress.2.1 -a 172.19.192.238
casAuthenPort.2.1 -i 2095 casAcctPort.2.1 -i 2096 casKey.2.1 -o king
casAddress.2.1 = 172.19.192.238
casAuthenPort.2.1 = 2095
casAcctPort.2.1 = 2096
casKey.2.1 = king
aaa-server5:/users/smetri>

```

After the Set Operation

After the above SNMP set operation, the configurations on the device change. The following output shows the output after the set operation.

```

Device# show running-config | include radius-server host

radius-server host 172.19.192.238 auth-port 1645 acct-port 1646
! The following line shows a change in the key value to "king."
radius-server host 172.19.192.238 auth-port 2095 acct-port 2096 key king

Device# show aaa servers

RADIUS: id 3, priority 1, host 172.19.192.238, auth-port 1645, acct-port 1646
State: current UP, duration 189s, previous duration 0s
  Dead: total time 0s, count 2
Authen: request 8, timeouts 8
  Response: unexpected 0, server error 0, incorrect 0, time 0ms
  Transaction: success 0, failure 4
Author: request 0, timeouts 0
  Response: unexpected 0, server error 0, incorrect 0, time 0ms
  Transaction: success 0, failure 0
Account: request 0, timeouts 0
  Response: unexpected 0, server error 0, incorrect 0, time 0ms
  Transaction: success 0, failure 0
Elapsed time since counters last cleared: 6m

```



```

! The following line shows a new server with new statistics.
RADIUS: id 4, priority 2, host 172.19.192.238, auth-port 2095, acct-port 2096
State: current UP, duration 209s, previous duration 0s
    Dead: total time 0s, count 7
Authen: request 0, timeouts 0
    Response: unexpected 0, server error 0, incorrect 0, time 0ms
    Transaction: success 0, failure 0
Author: request 0, timeouts 0
    Response: unexpected 0, server error 0, incorrect 0, time 0ms
    Transaction: success 0, failure 0
Account: request 0, timeouts 0
    Response: unexpected 0, server error 0, incorrect 0, time 0ms

```

Additional References for AAA-SERVER-MIB Set Operation

The following sections provide references related to the AAA-SERVER-MIB Set Operation feature.

Technical Assistance

Description	Link
<p>The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies.</p> <p>To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds.</p> <p>Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.</p>	<p>http://www.cisco.com/cisco/web/support/index.html</p>

Feature Information for AAA-SERVER-MIB Set Operation

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Table 127: Feature Information for AAA-SERVER-MIB Set Operation

Feature Name	Releases	Feature Information
AAA-SERVER-MIB Set Operation	Cisco IOS 15.2(1)E	<p>The AAA-SERVER-MIB Set Operation feature allows the authentication, authorization, and accounting (AAA) server configuration to be extended or expanded by using the CISCO-AAA-SERVER-MIB to create and add new AAA servers, modify the “KEY” under the CISCO-AAA-SERVER-MIB, and delete the AAA server configuration.</p> <p>The following commands were introduced or modified: show aaa servers, show running-config, show running-config vrf.</p>



CHAPTER 62

Configuring Secure Shell

The Secure Shell (SSH) feature is an application and a protocol that provides a secure replacement to the Berkeley r-tools. The protocol secures sessions using standard cryptographic mechanisms, and the application can be used similarly to the Berkeley rexec and rsh tools. Two versions of SSH are available: SSH Version 1 and SSH Version 2.

- [Prerequisites for Configuring Secure Shell, on page 1225](#)
- [Restrictions for Configuring Secure Shell, on page 1226](#)
- [Information About Configuring Secure Shell , on page 1226](#)
- [How to Configure Secure Shell, on page 1229](#)
- [Configuration Examples for Secure Shell, on page 1240](#)
- [Additional References for Secure Shell, on page 1242](#)
- [Feature Information for Configuring Secure Shell, on page 1242](#)

Prerequisites for Configuring Secure Shell

The following are the prerequisites for configuring the switch for secure shell (SSH):

- For SSH to work, the switch needs an Rivest, Shamir, and Adleman (RSA) public/private key pair. This is the same with Secure Copy Protocol (SCP), which relies on SSH for its secure transport.
- Before enabling SCP, you must correctly configure SSH, authentication, and authorization on the switch.
- Because SCP relies on SSH for its secure transport, the router must have an Rivest, Shamir, and Adelman (RSA) key pair.
- SCP relies on SSH for security.
- SCP requires that authentication, authorization, and accounting (AAA) authorization be configured so the router can determine whether the user has the correct privilege level.
- A user must have appropriate authorization to use SCP.
- A user who has appropriate authorization can use SCP to copy any file in the Cisco IOS File System (IFS) to and from a switch by using the **copy** command. An authorized administrator can also do this from a workstation.
- The Secure Shell (SSH) server requires an IPsec (Data Encryption Standard [DES] or 3DES) encryption software image; the SSH client requires an IPsec (DES or 3DES) encryption software image.)

- Configure a hostname and host domain for your device by using the **hostname** and **ip domain-name** commands in global configuration mode.

Restrictions for Configuring Secure Shell

The following are restrictions for configuring the Device for secure shell.

- The switch supports Rivest, Shamir, and Adelman (RSA) authentication.
- SSH supports only the execution-shell application.
- The SSH server and the SSH client are supported only on Data Encryption Standard (DES) (56-bit) and 3DES (168-bit) data encryption software. In DES software images, DES is the only encryption algorithm available. In 3DES software images, both DES and 3DES encryption algorithms are available.
- The Device supports the Advanced Encryption Standard (AES) encryption algorithm with a 128-bit key, 192-bit key, or 256-bit key. However, symmetric cipher AES to encrypt the keys is not supported.
- When using SCP, you cannot enter the password into the **copy** command. You must enter the password when prompted.
- The login banner is not supported in Secure Shell Version 1. It is supported in Secure Shell Version 2.
- The **-l** keyword and **userid** : {number} {ip-address} delimiter and arguments are mandatory when configuring the alternative method of Reverse SSH for console access.

Information About Configuring Secure Shell

Secure Shell (SSH) is a protocol that provides a secure, remote connection to a device. SSH provides more security for remote connections than Telnet does by providing strong encryption when a device is authenticated. This software release supports SSH Version 1 (SSHv1) and SSH Version 2 (SSHv2).

SSH and Switch Access

Secure Shell (SSH) is a protocol that provides a secure, remote connection to a device. SSH provides more security for remote connections than Telnet does by providing strong encryption when a device is authenticated. This software release supports SSH Version 1 (SSHv1) and SSH Version 2 (SSHv2).

SSH functions the same in IPv6 as in IPv4. For IPv6, SSH supports IPv6 addresses and enables secure, encrypted connections with remote IPv6 nodes over an IPv6 transport.

SSH Servers, Integrated Clients, and Supported Versions

The Secure Shell (SSH) Integrated Client feature is an application that runs over the SSH protocol to provide device authentication and encryption. The SSH client enables a Cisco device to make a secure, encrypted connection to another Cisco device or to any other device running the SSH server. This connection provides functionality similar to that of an outbound Telnet connection except that the connection is encrypted. With authentication and encryption, the SSH client allows for secure communication over an unsecured network.

The SSH server and SSH integrated client are applications that run on the switch. The SSH server works with the SSH client supported in this release and with non-Cisco SSH clients. The SSH client works with publicly and commercially available SSH servers. The SSH client supports the ciphers of Data Encryption Standard (DES), 3DES, and password authentication.

The switch supports an SSHv1 or an SSHv2 server.

The switch supports an SSHv1 client.



Note The SSH client functionality is available only when the SSH server is enabled.

User authentication is performed like that in the Telnet session to the device. SSH also supports the following user authentication methods:

- TACACS+
- RADIUS
- Local authentication and authorization

RSA Authentication Support

Rivest, Shamir, and Adleman (RSA) authentication available in Secure Shell (SSH) clients is not supported on the SSH server for Cisco software by default.

SSL Configuration Guidelines

When SSL is used in a switch cluster, the SSL session terminates at the cluster commander. Cluster member switches must run standard HTTP.

Before you configure a CA trustpoint, you should ensure that the system clock is set. If the clock is not set, the certificate is rejected due to an incorrect date.

In a switch stack, the SSL session terminates at the active switch.

Secure Copy Protocol Overview

The Secure Copy Protocol (SCP) feature provides a secure and authenticated method for copying switch configurations or switch image files. SCP relies on Secure Shell (SSH), an application and a protocol that provides a secure replacement for the Berkeley r-tools.

For SSH to work, the switch needs an RSA public/private key pair. This is the same with SCP, which relies on SSH for its secure transport.

Because SSH also relies on AAA authentication, and SCP relies further on AAA authorization, correct configuration is necessary.

- Before enabling SCP, you must correctly configure SSH, authentication, and authorization on the switch.
- Because SCP relies on SSH for its secure transport, the router must have an Rivest, Shamir, and Adelman (RSA) key pair.



Note When using SCP, you cannot enter the password into the copy command. You must enter the password when prompted.

Secure Copy Protocol

Isn't Secure Copy Protocol related closely enough to SSH that it could be used in this book? I have moved all of the item in this topic to prerequisites or restrictions.

The Secure Copy Protocol (SCP) feature provides a secure and authenticated method for copying device configurations or switch image files. The behavior of SCP is similar to that of remote copy (rcp), which comes from the Berkeley r-tools suite, except that SCP relies on SSH for security. SCP also requires that authentication, authorization, and accounting (AAA) authorization be configured so the device can determine whether the user has the correct privilege level. To configure the Secure Copy feature, you should understand the SCP concepts.

How Secure Copy Works

The behavior of Secure Copy (SCP) is similar to that of remote copy (RCP), which comes from the Berkeley r-tools suite (Berkeley university's own set of networking applications), except that SCP relies on Secure Shell (SSH) for security. In addition, SCP requires that authentication, authorization, and accounting (AAA) authorization be configured so that the device can determine whether the user has the correct privilege level.

SCP allows a user only with a privilege level of 15 to copy any file that exists in the Cisco IOS File System (IFS) to and from a device by using the **copy** command. An authorized administrator may also perform this action from a workstation.



Note Enable the SCP option while using the pscp.exe file with the Cisco software.

Reverse Telnet

Reverse telnet allows you to telnet to a certain port range and connect to terminal or auxiliary lines. Reverse telnet has often been used to connect a Cisco device that has many terminal lines to the consoles of other Cisco devices. Telnet makes it easy to reach the device console from anywhere simply by telnet to the terminal server on a specific line. This telnet approach can be used to configure a device even if all network connectivity to that device is disconnected. Reverse telnet also allows modems that are attached to Cisco devices to be used for dial-out (usually with a rotary device).

Reverse SSH

Reverse telnet can be accomplished using SSH. Unlike reverse telnet, SSH provides for secure connections. The Reverse SSH Enhancements feature provides you with a simplified method of configuring SSH. Using this feature, you no longer have to configure a separate line for every terminal or auxiliary line on which you want to enable SSH. The previous method of configuring reverse SSH limited the number of ports that can be accessed to 100. The Reverse SSH Enhancements feature removes the port number limitation.

How to Configure Secure Shell

Setting Up the Device to Run SSH

Follow these steps to set up your Device to run SSH:

Before you begin

Configure user authentication for local or remote access. This step is required. For more information, see Related Topics below.

SUMMARY STEPS

1. `enable`
2. `configure terminal`
3. `hostname hostname`
4. `ip domain-name domain_name`
5. `crypto key generate rsa`
6. `end`
7. `show running-config`
8. `copy running-config startup-config`

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> <code>enable</code>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode.
Step 3	hostname hostname Example: Device(config)# <code>hostname your_hostname</code>	Configures a hostname and IP domain name for your Device. Note Follow this procedure only if you are configuring the Device as an SSH server.
Step 4	ip domain-name domain_name Example:	Configures a host domain for your Device.

	Command or Action	Purpose
	Device(config)# <code>ip domain-name your_domain</code>	
Step 5	crypto key generate rsa Example: Device(config)# <code>crypto key generate rsa</code>	Enables the SSH server for local and remote authentication on the Device and generates an RSA key pair. Generating an RSA key pair for the Device automatically enables SSH. We recommend that a minimum modulus size of 1024 bits. When you generate RSA keys, you are prompted to enter a modulus length. A longer modulus length might be more secure, but it takes longer to generate and to use. Note Follow this procedure only if you are configuring the Device as an SSH server.
Step 6	end Example: Device(config)# <code>end</code>	Returns to privileged EXEC mode.
Step 7	show running-config Example: Device# <code>show running-config</code>	Verifies your entries.
Step 8	copy running-config startup-config Example: Device# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Configuring the SSH Server

Follow these steps to configure the SSH server:



Note This procedure is only required if you are configuring the Device as an SSH server.

SUMMARY STEPS

1. `enable`
2. `configure terminal`
3. `ip ssh version [1 | 2]`
4. `ip ssh {time-out seconds | authentication-retries number}`

5. Use one or both of the following:
 - `line vtyline_number[ending_line_number]`
 - `transport input ssh`
6. `end`
7. `show running-config`
8. `copy running-config startup-config`

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> <code>enable</code>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode.
Step 3	ip ssh version [1 2] Example: Device(config)# <code>ip ssh version 1</code>	(Optional) Configures the Device to run SSH Version 1 or SSH Version 2. <ul style="list-style-type: none"> • 1—Configure the Device to run SSH Version 1. • 2—Configure the Device to run SSH Version 2. If you do not enter this command or do not specify a keyword, the SSH server selects the latest SSH version supported by the SSH client. For example, if the SSH client supports SSHv1 and SSHv2, the SSH server selects SSHv2.
Step 4	ip ssh {time-out seconds authentication-retries number} Example: Device(config)# <code>ip ssh time-out 90</code> OR Device(config)# <code>ip ssh authentication-retries 2</code>	Configures the SSH control parameters: <ul style="list-style-type: none"> • time-out seconds: Specify the time-out value in seconds; the default is 120 seconds. The range is 0 to 120 seconds. This parameter applies to the SSH negotiation phase. After the connection is established, the Device uses the default time-out values of the CLI-based sessions. By default, up to five simultaneous, encrypted SSH connections for multiple CLI-based sessions over the network are available (session 0 to session 4). After the execution shell starts, the CLI-based session time-out value returns to the default of 10 minutes.

	Command or Action	Purpose
		<ul style="list-style-type: none"> • authentication-retries <i>number</i>: Specify the number of times that a client can re-authenticate to the server. The default is 3; the range is 0 to 5. <p>Repeat this step when configuring both parameters.</p>
Step 5	<p>Use one or both of the following:</p> <ul style="list-style-type: none"> • <code>line vtyline_number[ending_line_number]</code> • transport input ssh <p>Example:</p> <pre>Device(config)# line vty 1 10</pre> <p>or</p> <pre>Device(config-line)# transport input ssh</pre>	<p>(Optional) Configures the virtual terminal line settings.</p> <ul style="list-style-type: none"> • Enters line configuration mode to configure the virtual terminal line settings. For <i>line_number</i> and <i>ending_line_number</i>, specify a pair of lines. The range is 0 to 15. • Specifies that the Device prevent non-SSH Telnet connections. This limits the router to only SSH connections.
Step 6	<p>end</p> <p>Example:</p> <pre>Device(config-line)# end</pre>	Returns to privileged EXEC mode.
Step 7	<p>show running-config</p> <p>Example:</p> <pre>Device# show running-config</pre>	Verifies your entries.
Step 8	<p>copy running-config startup-config</p> <p>Example:</p> <pre>Device# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Invoking an SSH Client

Perform this task to invoke the Secure Shell (SSH) client. The SSH client runs in user EXEC mode and has no specific configuration tasks.

SUMMARY STEPS

1. **enable**
2. **ssh -l** *username* **-vrf** *vrf-name* *ip-address*

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	ssh -l <i>username</i> -vrf <i>vrf-name</i> <i>ip-address</i> Example: Device# ssh -l user1 -vrf vrf1 192.0.2.1	Invokes the SSH client to connect to an IP host or address in the specified virtual routing and forwarding (VRF) instance.

Troubleshooting Tips

- If your Secure Shell (SSH) configuration commands are rejected as illegal commands, you have not successfully generated an Rivest, Shamir, and Adleman (RSA) key pair for your device. Make sure that you have specified a hostname and domain. Then use the **crypto key generate rsa** command to generate an RSA key pair and enable the SSH server.
- When configuring the RSA key pair, you might encounter the following error messages:
 - No hostname specified.
You must configure a hostname for the device using the **hostname** global configuration command.
 - No domain specified.
You must configure a host domain for the device using the **ip domain-name** global configuration command.
- The number of allowable SSH connections is limited to the maximum number of vtys configured for the device. Each SSH connection uses a vty resource.
- SSH uses either local security or the security protocol that is configured through AAA on your device for user authentication. When configuring Authentication, Authorization, and Accounting (AAA), you must ensure that AAA is disabled on the console for user authentication. AAA authorization is disabled on the console by default. If AAA authorization is enabled on the console, disable it by configuring the **no aaa authorization console** command during the AAA configuration stage.

Configuring Reverse SSH for Console Access

To configure reverse SSH console access on the SSH server, perform the following steps.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **line *line-number* *ending-line-number***
4. **no exec**

5. **login authentication** *listname*
6. **transport input ssh**
7. **exit**
8. **exit**
9. **ssh -l** *userid* : {*number*} {*ip-address*}

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	line <i>line-number</i> <i>ending-line-number</i> Example: Device# line 1 3	Identifies a line for configuration and enters line configuration mode.
Step 4	no exec Example: Device(config-line)# no exec	Disables EXEC processing on a line.
Step 5	login authentication <i>listname</i> Example: Device(config-line)# login authentication default	Defines a login authentication mechanism for the lines. Note The authentication method must use a username and password.
Step 6	transport input ssh Example: Device(config-line)# transport input ssh	Defines which protocols to use to connect to a specific line of the device. <ul style="list-style-type: none"> • The ssh keyword must be used for the Reverse SSH Enhancements feature.
Step 7	exit Example: Device(config-line)# exit	Exits line configuration mode.
Step 8	exit Example:	Exits global configuration mode.

	Command or Action	Purpose
	Device(config)# exit	
Step 9	ssh -l <i>userid</i> : {<i>number</i>} {<i>ip-address</i>} Example: Device# ssh -l lab:1 router.example.com	Specifies the user ID to use when logging in on the remote networking device that is running the SSH server. <ul style="list-style-type: none"> • <i>userid</i> --User ID. • : --Signifies that a port number and terminal IP address will follow the <i>userid</i> argument. • <i>number</i> --Terminal or auxiliary line number. • <i>ip-address</i> --Terminal server IP address. Note The <i>userid</i> argument and :rotary { <i>number</i> }{ <i>ip-address</i> } delimiter and arguments are mandatory when configuring the alternative method of Reverse SSH for modem access.

Configuring Reverse SSH for Modem Access

In this configuration, reverse SSH is being configured on a modem used for dial-out lines. To get any of the dial-out modems, you can use any SSH client and start a SSH session as shown (in Step 10) to get to the next available modem from the rotary device.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **line *line-number* *ending-line-number***
4. **no exec**
5. **login authentication *listname***
6. **rotary *group***
7. **transport input ssh**
8. **exit**
9. **exit**
10. **ssh -l *userid* :rotary {*number*} {*ip-address*}**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.

	Command or Action	Purpose
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	line line-number ending-line-number Example: Device# line 1 200	Identifies a line for configuration and enters line configuration mode.
Step 4	no exec Example: Device(config-line)# no exec	Disables EXEC processing on a line.
Step 5	login authentication listname Example: Device(config-line)# login authentication default	Defines a login authentication mechanism for the lines. Note The authentication method must use a username and password.
Step 6	rotary group Example: Device(config-line)# rotary 1	Defines a group of lines consisting of one or more virtual terminal lines or one auxiliary port line.
Step 7	transport input ssh Example: Device(config-line)# transport input ssh	Defines which protocols to use to connect to a specific line of the device. <ul style="list-style-type: none">• The ssh keyword must be used for the Reverse SSH Enhancements feature.
Step 8	exit Example: Device(config-line)# exit	Exits line configuration mode.
Step 9	exit Example: Device(config)# exit	Exits global configuration mode.
Step 10	ssh -l userid :rotary {number} {ip-address} Example: Device# ssh -l lab:rotary1 router.example.com	Specifies the user ID to use when logging in on the remote networking device that is running the SSH server. <ul style="list-style-type: none">• <i>userid</i> --User ID.• : --Signifies that a port number and terminal IP address will follow the <i>userid</i> argument.

	Command or Action	Purpose
		<ul style="list-style-type: none"> • <i>number</i> --Terminal or auxiliary line number. • <i>ip-address</i> --Terminal server IP address. <p>Note The <i>userid</i> argument and :rotary{<i>number</i>} {<i>ip-address</i>} delimiter and arguments are mandatory when configuring the alternative method of Reverse SSH for modem access.</p>

Troubleshooting Reverse SSH on the Client

To troubleshoot the reverse SSH configuration on the client (remote device), perform the following steps.

SUMMARY STEPS

1. `enable`
2. `debug ip ssh client`

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Device> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	debug ip ssh client Example: <pre>Device# debug ip ssh client</pre>	Displays debugging messages for the SSH client.

Troubleshooting Reverse SSH on the Server

To troubleshoot the reverse SSH configuration on the terminal server, perform the following steps. The steps may be configured in any order or independent of one another.

SUMMARY STEPS

1. `enable`
2. `debug ip ssh`
3. `show ssh`
4. `show line`

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	debug ip ssh Example: Device# debug ip ssh	Displays debugging messages for the SSH server.
Step 3	show ssh Example: Device# show ssh	Displays the status of the SSH server connections.
Step 4	show line Example: Device# show line	Displays parameters of a terminal line.

Monitoring the SSH Configuration and Status

This table displays the SSH server configuration and status.

Table 128: Commands for Displaying the SSH Server Configuration and Status

Command	Purpose
show ip ssh	Shows the version and configuration information for the SSH server.
show ssh	Shows the status of the SSH server.

Configuring Secure Copy

To configure a Cisco device for Secure Copy (SCP) server-side functionality, perform the following steps.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **aaa new-model**
4. **aaa authentication login** {default | list-name} method1 [method2...]
5. **aaa authorization** {network | exec | commands level | reverse-access | configuration} {default | list-name} [method1 [method2...]]

6. **username** *name* [**privilege level**] **password** *encryption-type encrypted-password*
7. **ip scp server enable**
8. **exit**
9. **show running-config**
10. **debug ip scp**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	aaa new-model Example: Device(config)# aaa new-model	Sets AAA authentication at login.
Step 4	aaa authentication login { default <i>list-name</i> } <i>method1</i> [<i>method2...</i>] Example: Device(config)# aaa authentication login default group tacacs+	Enables the AAA access control system.
Step 5	aaa authorization { network exec commands level reverse-access configuration } { default <i>list-name</i> } [<i>method1</i> [<i>method2...</i>]] Example: Device(config)# aaa authorization exec default group tacacs+	Sets parameters that restrict user access to a network. Note The exec keyword runs authorization to determine if the user is allowed to run an EXEC shell; therefore, you must use the exec keyword when you configure SCP.
Step 6	username <i>name</i> [privilege level] password <i>encryption-type encrypted-password</i> Example: Device(config)# username superuser privilege 2 password 0 superpassword	Establishes a username-based authentication system. Note You may omit this step if a network-based authentication mechanism, such as TACACS+ or RADIUS, has been configured.
Step 7	ip scp server enable Example:	Enables SCP server-side functionality.

	Command or Action	Purpose
	Device(config)# ip scp server enable	
Step 8	exit Example: Device(config)# exit	Exits global configuration mode and returns to privileged EXEC mode.
Step 9	show running-config Example: Device# show running-config	(Optional) Displays the SCP server-side functionality.
Step 10	debug ip scp Example: Device# debug ip scp	(Optional) Troubleshoots SCP authentication problems.

Configuration Examples for Secure Shell

Example: Secure Copy Configuration Using Local Authentication

The following example shows how to configure the server-side functionality of Secure Copy (SCP). This example uses a locally defined username and password.

```
! AAA authentication and authorization must be configured properly in order for SCP to work.
aaa new-model
aaa authentication login default local
aaa authorization exec default local
username user1 privilege 15 password 0 lab
! SSH must be configured and functioning properly.
ip scp server enable
```

Example: SCP Server-Side Configuration Using Network-Based Authentication

The following example shows how to configure the server-side functionality of SCP using a network-based authentication mechanism:

```
! AAA authentication and authorization must be configured properly for SCP to work.
aaa new-model
aaa authentication login default group tacacs+
aaa authorization exec default group tacacs+
! SSH must be configured and functioning properly.
ip ssh time-out 120
ip ssh authentication-retries 3
ip scp server enable
```

Example Reverse SSH Console Access

The following configuration example shows that reverse SSH has been configured for console access for terminal lines 1 through 3:

Terminal Server Configuration

```
line 1 3
  no exec
  login authentication default
  transport input ssh
```

Client Configuration

The following commands configured on the SSH client will form the reverse SSH session with lines 1, 2, and 3, respectively:

```
ssh -l lab:1 router.example.com
ssh -l lab:2 router.example.com
ssh -l lab:3 router.example.com
```

Example Reverse SSH Modem Access

The following configuration example shows that dial-out lines 1 through 200 have been grouped under rotary group 1 for modem access:

```
line 1 200
  no exec
  login authentication default
  rotary 1
  transport input ssh
  exit
```

The following command shows that reverse SSH will connect to the first free line in the rotary group:

```
ssh -l lab:rotary1 router.example.com
```

Example: Monitoring the SSH Configuration and Status

To verify that the Secure Shell (SSH) server is enabled and to display the version and configuration data for your SSH connection, use the **show ip ssh** command. The following example shows that SSH is enabled:

```
Device# show ip ssh

SSH Enabled - version 1.5
Authentication timeout: 120 secs; Authentication retries: 3
```

The following example shows that SSH is disabled:

```
Device# show ip ssh

%SSH has not been enabled
```

To verify the status of your SSH server connections, use the **show ssh** command. The following example shows the SSH server connections on the device when SSH is enabled:

```
Device# show ssh

Connection      Version      Encryption State Username
0 1.5 3DES Session Started guest
```

The following example shows that SSH is disabled:

```
Device# show ssh

%No SSH server connections running.
```

Additional References for Secure Shell

Technical Assistance

Description	Link
<p>The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies.</p> <p>To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds.</p> <p>Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.</p>	http://www.cisco.com/support

Feature Information for Configuring Secure Shell

Release	Feature Information
Cisco IOS Release 15.0(2)EX1	This feature was introduced.
Cisco IOS Release 15.2(5)E	Note Starting with Cisco IOS Release 15.2(5)E, Secure Shell Version 1 (SSHv1) is deprecated.

Release	Feature Information
Cisco IOS 15.2(1)E	<p>The Reverse SSH Enhancements feature, which is supported for SSH Version 1 and 2, provides an alternative way to configure reverse Secure Shell (SSH) so that separate lines do not need to be configured for every terminal or auxiliary line on which SSH must be enabled. This feature also eliminates the rotary-group limitation.</p> <p>This feature was supported on CAT4500-X, CAT4500E-SUP6E, CAT4500E-SUP6L-E, CAT4500E-SUP7E, CAT4500E-SUP7L-E.</p> <p>The following command was introduced: ssh.</p>



CHAPTER 63

Secure Shell Version 2 Support

The Secure Shell Version 2 Support feature allows you to configure Secure Shell (SSH) Version 2. (SSH Version 1 support was implemented in an earlier Cisco software release.) SSH runs on top of a reliable transport layer and provides strong authentication and encryption capabilities. The only reliable transport that is defined for SSH is TCP. SSH provides a means to securely access and securely execute commands on another computer over a network. The Secure Copy Protocol (SCP) feature that is provided with SSH allows for the secure transfer of files.

- [Information About Secure Shell Version 2 Support, on page 1245](#)
- [How to Configure Secure Shell Version 2 Support, on page 1248](#)
- [Configuration Examples for Secure Shell Version 2 Support, on page 1263](#)
- [Additional References for Secure Shell Version 2 Support, on page 1268](#)
- [Feature Information for Secure Shell Version 2 Support, on page 1268](#)

Information About Secure Shell Version 2 Support

Secure Shell Version 2

The Secure Shell Version 2 Support feature allows you to configure SSH Version 2.

The configuration for the SSH Version 2 server is similar to the configuration for SSH Version 1. The **ip ssh version** command defines the SSH version to be configured. If you do not configure this command, SSH by default runs in compatibility mode; that is, both SSH Version 1 and SSH Version 2 connections are honored.



Note SSH Version 1 is a protocol that has never been defined in a standard. If you do not want your device to fall back to the undefined protocol (Version 1), you should use the **ip ssh version** command and specify Version 2.

The **ip ssh rsa keypair-name** command enables an SSH connection using the Rivest, Shamir, and Adleman (RSA) keys that you have configured. Previously, SSH was linked to the first RSA keys that were generated (that is, SSH was enabled when the first RSA key pair was generated). This behavior still exists, but by using the **ip ssh rsa keypair-name** command, you can overcome this behavior. If you configure the **ip ssh rsa keypair-name** command with a key pair name, SSH is enabled if the key pair exists or SSH will be enabled if the key pair is generated later. If you use this command to enable SSH, you are not forced to configure a hostname and a domain name, which was required in SSH Version 1 of the Cisco software.



Note The login banner is supported in SSH Version 2, but it is not supported in Secure Shell Version 1.

Secure Shell Version 2 Enhancements

The SSH Version 2 Enhancements feature includes a number of additional capabilities such as supporting Virtual Routing and Forwarding (VRF)-Aware SSH, SSH debug enhancements, and Diffie-Hellman (DH) group exchange support.



Note The VRF-Aware SSH feature is supported depending on your release.

The Cisco SSH implementation has traditionally used 768-bit modulus, but with an increasing need for higher key sizes to accommodate DH Group 14 (2048 bits) and Group 16 (4096 bits) cryptographic applications, a message exchange between the client and the server to establish the favored DH group becomes necessary. The **ip ssh dh min size** command configures the modulus size on the SSH server. In addition to this, the **ssh** command was extended to add VRF awareness to the SSH client-side functionality through which the VRF instance name in the client is provided with the IP address to look up the correct routing table and establish a connection.

Debugging was enhanced by modifying SSH debug commands. The **debug ip ssh** command was extended to simplify the debugging process. Before the simplification of the debugging process, this command printed all debug messages related to SSH regardless of what was specifically required. The behavior still exists, but if you configure the **debug ip ssh** command with a keyword, messages are limited to information specified by the keyword.

Secure Shell Version 2 Enhancements for RSA Keys

Cisco SSH Version 2 supports keyboard-interactive and password-based authentication methods. The SSH Version 2 Enhancements for RSA Keys feature also supports RSA-based public key authentication for the client and the server.

User authentication—RSA-based user authentication uses a private/public key pair associated with each user for authentication. The user must generate a private/public key pair on the client and configure a public key on the Cisco SSH server to complete the authentication.

An SSH user trying to establish credentials provides an encrypted signature using the private key. The signature and the user's public key are sent to the SSH server for authentication. The SSH server computes a hash over the public key provided by the user. The hash is used to determine if the server has a matching entry. If a match is found, an RSA-based message verification is performed using the public key. Hence, the user is authenticated or denied access based on the encrypted signature.

Server authentication—While establishing an SSH session, the Cisco SSH client authenticates the SSH server by using the server host keys available during the key exchange phase. SSH server keys are used to identify the SSH server. These keys are created at the time of enabling SSH and must be configured on the client.

For server authentication, the Cisco SSH client must assign a host key for each server. When the client tries to establish an SSH session with a server, the client receives the signature of the server as part of the key exchange message. If the strict host key checking flag is enabled on the client, the client checks if it has the host key entry corresponding to the server. If a match is found, the client tries to validate the signature by

using the server host key. If the server is successfully authenticated, the session establishment continues; otherwise, it is terminated and displays a “Server Authentication Failed” message.



Note Storing public keys on a server uses memory; therefore, the number of public keys configurable on an SSH server is restricted to ten users, with a maximum of two public keys per user.



Note RSA-based user authentication is supported by the Cisco server, but Cisco clients cannot propose public key as an authentication method. If the Cisco server receives a request from an open SSH client for RSA-based authentication, the server accepts the authentication request.



Note For server authentication, configure the RSA public key of the server manually and configure the **ip ssh stricthostkeycheck** command on the Cisco SSH client.

SNMP Trap Generation

Depending on your release, Simple Network Management Protocol (SNMP) traps are generated automatically when an SSH session terminates if the traps have been enabled and SNMP debugging has been enabled. For information about enabling SNMP traps, see the “Configuring SNMP Support” module in the *SNMP Configuration Guide*.



Note When you configure the **snmp-server host** command, the IP address must be the address of the PC that has the SSH (telnet) client and that has IP connectivity to the SSH server.

You must also enable SNMP debugging using the **debug snmp packet** command to display the traps. The trap information includes information such as the number of bytes sent and the protocol that was used for the SSH session.

The following example shows that an SNMP trap is set. The trap notification is generated automatically when the SSH session terminates. In the example, a.b.c.d is the IP address of the SSH client.

```
snmp-server
snmp-server host a.b.c.d public tty
```

The following is sample output from the **debug snmp packet** command. The output provides SNMP trap information for an SSH session.

```
Switch# debug snmp packet

SNMP packet debugging is on
Device1# ssh -l lab 10.0.0.2
Password:

Switch# exit
```

```
[Connection to 10.0.0.2 closed by foreign host]
Device1#
*Jul 18 10:18:42.619: SNMP: Queuing packet to 10.0.0.2
*Jul 18 10:18:42.619: SNMP: V1 Trap, ent cisco, addr 10.0.0.1, gentrap 6, spectrap 1
local.9.3.1.1.2.1 = 6
tcpConnEntry.1.10.0.0.1.22.10.0.0.2.55246 = 4
ltcpConnEntry.5.10.0.0.1.22.10.0.0.2.55246 = 1015
ltcpConnEntry.1.10.0.0.1.22.10.0.0.2.55246 = 1056
ltcpConnEntry.2.10.0.0.1.22.10.0.0.2.55246 = 1392
local.9.2.1.18.2 = lab
*Jul 18 10:18:42.879: SNMP: Packet sent via UDP to 10.0.0.2

Switch#
```

SSH Keyboard Interactive Authentication

The SSH Keyboard Interactive Authentication feature, also known as Generic Message Authentication for SSH, is a method that can be used to implement different types of authentication mechanisms. Basically, any currently supported authentication method that requires only user input can be performed with this feature. The feature is automatically enabled.

The following methods are supported:

- Password
- SecurID and hardware tokens printing a number or a string in response to a challenge sent by the server
- Pluggable Authentication Module (PAM)
- S/KEY (and other One-Time-Pads)

How to Configure Secure Shell Version 2 Support

Configuring a Device for SSH Version 2 Using a Hostname and Domain Name

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **hostname** *name*
4. **ip domain-name** *name*
5. **crypto key generate rsa**
6. **ip ssh** [**time-out** *seconds* | **authentication-retries** *integer*]
7. **ip ssh version** [**1** | **2**]
8. **exit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable	Enables privileged EXEC mode.

	Command or Action	Purpose
	Example: Device> enable	<ul style="list-style-type: none"> Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	hostname <i>name</i> Example: Device(config)# hostname cisco7200	Configures a hostname for your device.
Step 4	ip domain-name <i>name</i> Example: cisco7200(config)# ip domain-name example.com	Configures a domain name for your device.
Step 5	crypto key generate rsa Example: cisco7200(config)# crypto key generate rsa	Enables the SSH server for local and remote authentication.
Step 6	ip ssh [time-out <i>seconds</i> authentication-retries <i>integer</i>] Example: cisco7200(config)# ip ssh time-out 120	(Optional) Configures SSH control variables on your device.
Step 7	ip ssh version [1 2] Example: cisco7200(config)# ip ssh version 1	(Optional) Specifies the version of SSH to be run on your device.
Step 8	exit Example: cisco7200(config)# exit	Exits global configuration mode and enters privileged EXEC mode. <ul style="list-style-type: none"> Use no hostname command to return to the default host.

Configuring a Device for SSH Version 2 Using RSA Key Pairs

SUMMARY STEPS

- enable
- configure terminal
- ip ssh rsa keypair-name *keypair-name*

4. **crypto key generate rsa usage-keys label *key-label* modulus *modulus-size***
5. **ip ssh [time-out *seconds* | authentication-retries *integer*]**
6. **ip ssh version 2**
7. **exit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	ip ssh rsa keypair-name <i>keypair-name</i> Example: Device(config)# ip ssh rsa keypair-name sshkeys	Specifies the RSA key pair to be used for SSH. Note A Cisco device can have many RSA key pairs.
Step 4	crypto key generate rsa usage-keys label <i>key-label</i> modulus <i>modulus-size</i> Example: Device(config)# crypto key generate rsa usage-keys label sshkeys modulus 768	Enables the SSH server for local and remote authentication on the device. <ul style="list-style-type: none"> • For SSH Version 2, the modulus size must be at least 768 bits. Note To delete the RSA key pair, use the crypto key zeroize rsa command. When you delete the RSA key pair, you automatically disable the SSH server.
Step 5	ip ssh [time-out <i>seconds</i> authentication-retries <i>integer</i>] Example: Device(config)# ip ssh time-out 12	Configures SSH control variables on your device.
Step 6	ip ssh version 2 Example: Device(config)# ip ssh version 2	Specifies the version of SSH to be run on the device.
Step 7	exit Example: Device(config)# exit	Exits global configuration mode and enters privileged EXEC mode.

Configuring the Cisco SSH Server to Perform RSA-Based User Authentication

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **hostname** *name*
4. **ip domain-name** *name*
5. **crypto key generate rsa**
6. **ip ssh pubkey-chain**
7. **username** *username*
8. **key-string**
9. **key-hash** *key-type key-name*
10. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	hostname <i>name</i> Example: Device(config)# hostname host1	Specifies the hostname.
Step 4	ip domain-name <i>name</i> Example: host1(config)# ip domain-name name1	Defines a default domain name that the Cisco software uses to complete unqualified hostnames.
Step 5	crypto key generate rsa Example: host1(config)# crypto key generate rsa	Generates RSA key pairs.
Step 6	ip ssh pubkey-chain Example: host1(config)# ip ssh pubkey-chain	Configures SSH-RSA keys for user and server authentication on the SSH server and enters public-key configuration mode.

	Command or Action	Purpose
		<ul style="list-style-type: none"> The user authentication is successful if the RSA public key stored on the server is verified with the public or the private key pair stored on the client.
Step 7	username <i>username</i> Example: <pre>host1(conf-ssh-pubkey)# username user1</pre>	Configures the SSH username and enters public-key user configuration mode.
Step 8	key-string Example: <pre>host1(conf-ssh-pubkey-user)# key-string</pre>	Specifies the RSA public key of the remote peer and enters public-key data configuration mode. Note You can obtain the public key value from an open SSH client; that is, from the <code>.ssh/id_rsa.pub</code> file.
Step 9	key-hash <i>key-type key-name</i> Example: <pre>host1(conf-ssh-pubkey-data)# key-hash ssh-rsa key1</pre>	(Optional) Specifies the SSH key type and version. <ul style="list-style-type: none"> The key type must be <code>ssh-rsa</code> for the configuration of private public key pairs. This step is optional only if the key-string command is configured. You must configure either the key-string command or the key-hash command. Note You can use a hashing software to compute the hash of the public key string, or you can also copy the hash value from another Cisco device. Entering the public key data using the key-string command is the preferred way to enter the public key data for the first time.
Step 10	end Example: <pre>host1(conf-ssh-pubkey-data)# end</pre>	Exits public-key data configuration mode and returns to privileged EXEC mode. <ul style="list-style-type: none"> Use no hostname command to return to the default host.

Configuring the Cisco IOS SSH Client to Perform RSA-Based Server Authentication

SUMMARY STEPS

- enable**
- configure terminal**
- hostname** *name*

4. **ip domain-name** *name*
5. **crypto key generate rsa**
6. **ip ssh pubkey-chain**
7. **server** *server-name*
8. **key-string**
9. **exit**
10. **key-hash** *key-type key-name*
11. **end**
12. **configure terminal**
13. **ip ssh stricthostkeycheck**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	hostname <i>name</i> Example: Device(config)# hostname host1	Specifies the hostname.
Step 4	ip domain-name <i>name</i> Example: host1(config)# ip domain-name name1	Defines a default domain name that the Cisco software uses to complete unqualified hostnames.
Step 5	crypto key generate rsa Example: host1(config)# crypto key generate rsa	Generates RSA key pairs.
Step 6	ip ssh pubkey-chain Example: host1(config)# ip ssh pubkey-chain	Configures SSH-RSA keys for user and server authentication on the SSH server and enters public-key configuration mode.
Step 7	server <i>server-name</i> Example: host1(conf-ssh-pubkey)# server server1	Enables the SSH server for public-key authentication on the device and enters public-key server configuration mode.

	Command or Action	Purpose
Step 8	key-string Example: <pre>host1(conf-ssh-pubkey-server)# key-string</pre>	Specifies the RSA public-key of the remote peer and enters public key data configuration mode. Note You can obtain the public key value from an open SSH client; that is, from the .ssh/id_rsa.pub file.
Step 9	exit Example: <pre>host1(conf-ssh-pubkey-data)# exit</pre>	Exits public-key data configuration mode and enters public-key server configuration mode.
Step 10	key-hash <i>key-type key-name</i> Example: <pre>host1(conf-ssh-pubkey-server)# key-hash ssh-rsa key1</pre>	(Optional) Specifies the SSH key type and version. <ul style="list-style-type: none"> • The key type must be ssh-rsa for the configuration of private/public key pairs. • This step is optional only if the key-string command is configured. • You must configure either the key-string command or the key-hash command. Note You can use a hashing software to compute the hash of the public key string, or you can copy the hash value from another Cisco device. Entering the public key data using the key-string command is the preferred way to enter the public key data for the first time.
Step 11	end Example: <pre>host1(conf-ssh-pubkey-server)# end</pre>	Exits public-key server configuration mode and returns to privileged EXEC mode.
Step 12	configure terminal Example: <pre>host1# configure terminal</pre>	Enters global configuration mode.
Step 13	ip ssh stricthostkeycheck Example: <pre>host1(config)# ip ssh stricthostkeycheck</pre>	Ensures that server authentication takes place. <ul style="list-style-type: none"> • The connection is terminated in case of a failure. • Use no hostname command to return to the default host.

Starting an Encrypted Session with a Remote Device



Note The device with which you want to connect must support a Secure Shell (SSH) server that has an encryption algorithm that is supported in Cisco software. Also, you need not enable your device. SSH can be run in disabled mode.

SUMMARY STEPS

1. `ssh [-v {1 | 2}] [-c {aes128-ctr | aes192-ctr | aes256-ctr | aes128-cbc | 3des | aes192-cbc | aes256-cbc} | -l user-id | -l user-id:vrf-name number ip-address ip-address | -l user-id:rotary number ip-address | -m {hmac-md5-128 | hmac-md5-96 | hmac-sha1-160 | hmac-sha1-96} | -o numberofpasswordprompts n | -p port-num] {ip-addr | hostname} [command | -vrf]`

DETAILED STEPS

	Command or Action	Purpose
Step 1	<pre>ssh [-v {1 2}] [-c {aes128-ctr aes192-ctr aes256-ctr aes128-cbc 3des aes192-cbc aes256-cbc} -l <i>user-id</i> -l <i>user-id:vrf-name number ip-address ip-address</i> -l <i>user-id:rotary number ip-address</i> -m {hmac-md5-128 hmac-md5-96 hmac-sha1-160 hmac-sha1-96} -o <i>numberofpasswordprompts n</i> -p <i>port-num</i>] {<i>ip-addr</i> <i>hostname</i>} [<i>command</i> -vrf]</pre> <p>Example:</p> <pre>Device# ssh -v 2 -c aes256-ctr -m hmac-sha1-96 -l user2 10.76.82.24</pre>	Starts an encrypted session with a remote networking device.

Enabling Secure Copy Protocol on the SSH Server



Note The following task configures the server-side functionality for SCP. This task shows a typical configuration that allows the device to securely copy files from a remote workstation.

SUMMARY STEPS

1. `enable`
2. `configure terminal`
3. `aaa new-model`
4. `aaa authentication login default local`
5. `aaa authorization exec defaultlocal`
6. `username name privilege privilege-level password password`
7. `ip ssh time-out seconds`
8. `ip ssh authentication-retries integer`

9. `ip scpserverenable`
10. `exit`
11. `debug ip scp`

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Device> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: <pre>Device# configure terminal</pre>	Enters global configuration mode.
Step 3	aaa new-model Example: <pre>Device(config)# aaa new-model</pre>	Enables the AAA access control model.
Step 4	aaa authentication login default local Example: <pre>Device(config)# aaa authentication login default local</pre>	Sets AAA authentication at login to use the local username database for authentication.
Step 5	aaa authorization exec defaultlocal Example: <pre>Device(config)# aaa authorization exec default local</pre>	Sets the parameters that restrict user access to a network, runs the authorization to determine if the user ID is allowed to run an EXEC shell, and specifies that the system must use the local database for authorization.
Step 6	username name privilege privilege-level password password Example: <pre>Device(config)# username samplename privilege 15 password password1</pre>	Establishes a username-based authentication system, and specifies the username, privilege level, and an unencrypted password. Note The minimum value for the <i>privilege-level</i> argument is 15. A privilege level of less than 15 results in the connection closing.
Step 7	ip ssh time-out seconds Example: <pre>Device(config)# ip ssh time-out 120</pre>	Sets the time interval (in seconds) that the device waits for the SSH client to respond.
Step 8	ip ssh authentication-retries integer Example:	Sets the number of authentication attempts after which the interface is reset.

	Command or Action	Purpose
	Device(config)# ip ssh authentication-retries 3	
Step 9	ip scpserverenable Example: Device(config)# ip scp server enable	Enables the device to securely copy files from a remote workstation.
Step 10	exit Example: Device(config)# exit	Exits global configuration mode and returns to privileged EXEC mode.
Step 11	debug ip scp Example: Device# debug ip scp	(Optional) Provides diagnostic information about SCP authentication problems.

Verifying the Status of the Secure Shell Connection

SUMMARY STEPS

1. enable
2. show ssh
3. exit

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	show ssh Example: Device# show ssh	Displays the status of SSH server connections.
Step 3	exit Example: Device# exit	Exits privileged EXEC mode and returns to user EXEC mode.

Examples

The following sample output from the **show ssh** command displays status of various SSH Version 1 and Version 2 connections for Version 1 and Version 2 connections:

```
-----
Device# show ssh

Connection      Version Encryption      State                Username
0               1.5      3DES              Session started     lab
Connection Version Mode Encryption Hmac                State
Username
1               2.0      IN   aes128-cbc hmac-md5      Session started     lab
1               2.0      OUT  aes128-cbc hmac-md5      Session started     lab
-----
```

The following sample output from the **show ssh** command displays status of various SSH Version 1 and Version 2 connections for a Version 2 connection with no Version 1 connection:

```
-----
Device# show ssh

Connection Version Mode Encryption Hmac                State
Username
1               2.0      IN   aes128-cbc hmac-md5      Session started     lab
1               2.0      OUT  aes128-cbc hmac-md5      Session started     lab
%No SSHv1 server connections running.
-----
```

The following sample output from the **show ssh** command displays status of various SSH Version 1 and Version 2 connections for a Version 1 connection with no Version 2 connection:

```
-----
Device# show ssh

Connection      Version Encryption      State                Username
0               1.5      3DES              Session started     lab
%No SSHv2 server connections running.
-----
```

Verifying the Secure Shell Status

SUMMARY STEPS

1. enable
2. show ip ssh
3. exit

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable	Enables privileged EXEC mode.

	Command or Action	Purpose
	Example: Device> enable	<ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	show ip ssh Example: Device# show ip ssh	Displays the version and configuration data for SSH.
Step 3	exit Example: Device# exit	Exits privileged EXEC mode and returns to user EXEC mode.

Examples

The following sample output from the **show ip ssh** command displays the version of SSH that is enabled, the authentication timeout values, and the number of authentication retries for Version 1 and Version 2 connections:

```
-----
Device# show ip ssh

SSH Enabled - version 1.99
Authentication timeout: 120 secs; Authentication retries: 3
-----
```

The following sample output from the **show ip ssh** command displays the version of SSH that is enabled, the authentication timeout values, and the number of authentication retries for a Version 2 connection with no Version 1 connection:

```
-----
Device# show ip ssh

SSH Enabled - version 2.0
Authentication timeout: 120 secs; Authentication retries: 3
-----
```

The following sample output from the **show ip ssh** command displays the version of SSH that is enabled, the authentication timeout values, and the number of authentication retries for a Version 1 connection with no Version 2 connection:

```
-----
Device# show ip ssh

3d06h: %SYS-5-CONFIG_I: Configured from console by console
SSH Enabled - version 1.5
Authentication timeout: 120 secs; Authentication retries: 3
-----
```

Monitoring and Maintaining Secure Shell Version 2

SUMMARY STEPS

1. enable
2. debug ip ssh
3. debug snmp packet

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	debug ip ssh Example: Device# debug ip ssh	Enables debugging of SSH.
Step 3	debug snmp packet Example: Device# debug snmp packet	Enables debugging of every SNMP packet sent or received by the device.

Example

The following sample output from the **debug ip ssh** command shows the connection is an SSH Version 2 connection:

```
Device# debug ip ssh

00:33:55: SSH1: starting SSH control process
00:33:55: SSH1: sent protocol version id SSH-1.99-Cisco-1.25
00:33:55: SSH1: protocol version id is - SSH-2.0-OpenSSH_2.5.2p2
00:33:55: SSH2 1: send: len 280 (includes padlen 4)
00:33:55: SSH2 1: SSH2_MSG_KEXINIT sent
00:33:55: SSH2 1: ssh_receive: 536 bytes received
00:33:55: SSH2 1: input: packet len 632
00:33:55: SSH2 1: partial packet 8, need 624, maclen 0
00:33:55: SSH2 1: ssh_receive: 96 bytes received
00:33:55: SSH2 1: partial packet 8, need 624, maclen 0
00:33:55: SSH2 1: input: padlen 11
00:33:55: SSH2 1: received packet type 20
00:33:55: SSH2 1: SSH2_MSG_KEXINIT received
00:33:55: SSH2: kex: client->server aes128-cbc hmac-md5 none
00:33:55: SSH2: kex: server->client aes128-cbc hmac-md5 none
00:33:55: SSH2 1: expecting SSH2_MSG_KEXDH_INIT
00:33:55: SSH2 1: ssh_receive: 144 bytes received
00:33:55: SSH2 1: input: packet len 144
00:33:55: SSH2 1: partial packet 8, need 136, maclen 0
```

```
00:33:55: SSH2 1: input: padlen 5
00:33:55: SSH2 1: received packet type 30
00:33:55: SSH2 1: SSH2_MSG_KEXDH_INIT received
00:33:55: SSH2 1: signature length 111
00:33:55: SSH2 1: send: len 384 (includes padlen 7)
00:33:55: SSH2: kex_derive_keys complete
00:33:55: SSH2 1: send: len 16 (includes padlen 10)
00:33:55: SSH2 1: newkeys: mode 1
00:33:55: SSH2 1: SSH2_MSG_NEWKEYS sent
00:33:55: SSH2 1: waiting for SSH2_MSG_NEWKEYS
00:33:55: SSH2 1: ssh_receive: 16 bytes received
00:33:55: SSH2 1: input: packet len 16
00:33:55: SSH2 1: partial packet 8, need 8, maclen 0
00:33:55: SSH2 1: input: padlen 10
00:33:55: SSH2 1: newkeys: mode 0
00:33:55: SSH2 1: received packet type 2100:33:55: SSH2 1: SSH2_MSG_NEWKEYS received
00:33:56: SSH2 1: ssh_receive: 48 bytes received
00:33:56: SSH2 1: input: packet len 32
00:33:56: SSH2 1: partial packet 16, need 16, maclen 16
00:33:56: SSH2 1: MAC #3 ok
00:33:56: SSH2 1: input: padlen 10
00:33:56: SSH2 1: received packet type 5
00:33:56: SSH2 1: send: len 32 (includes padlen 10)
00:33:56: SSH2 1: done calc MAC out #3
00:33:56: SSH2 1: ssh_receive: 64 bytes received
00:33:56: SSH2 1: input: packet len 48
00:33:56: SSH2 1: partial packet 16, need 32, maclen 16
00:33:56: SSH2 1: MAC #4 ok
00:33:56: SSH2 1: input: padlen 9
00:33:56: SSH2 1: received packet type 50
00:33:56: SSH2 1: send: len 32 (includes padlen 13)
00:33:56: SSH2 1: done calc MAC out #4
00:34:04: SSH2 1: ssh_receive: 160 bytes received
00:34:04: SSH2 1: input: packet len 64
00:34:04: SSH2 1: partial packet 16, need 48, maclen 16
00:34:04: SSH2 1: MAC #5 ok
00:34:04: SSH2 1: input: padlen 13
00:34:04: SSH2 1: received packet type 50
00:34:04: SSH2 1: send: len 16 (includes padlen 10)
00:34:04: SSH2 1: done calc MAC out #5
00:34:04: SSH2 1: authentication successful for lab
00:34:04: SSH2 1: input: packet len 64
00:34:04: SSH2 1: partial packet 16, need 48, maclen 16
00:34:04: SSH2 1: MAC #6 ok
00:34:04: SSH2 1: input: padlen 6
00:34:04: SSH2 1: received packet type 2
00:34:04: SSH2 1: ssh_receive: 64 bytes received
00:34:04: SSH2 1: input: packet len 48
00:34:04: SSH2 1: partial packet 16, need 32, maclen 16
00:34:04: SSH2 1: MAC #7 ok
00:34:04: SSH2 1: input: padlen 19
00:34:04: SSH2 1: received packet type 90
00:34:04: SSH2 1: channel open request
00:34:04: SSH2 1: send: len 32 (includes padlen 10)
00:34:04: SSH2 1: done calc MAC out #6
00:34:04: SSH2 1: ssh_receive: 192 bytes received
00:34:04: SSH2 1: input: packet len 64
00:34:04: SSH2 1: partial packet 16, need 48, maclen 16
00:34:04: SSH2 1: MAC #8 ok
00:34:04: SSH2 1: input: padlen 13
00:34:04: SSH2 1: received packet type 98
00:34:04: SSH2 1: pty-req request
00:34:04: SSH2 1: setting TTY - requested: height 24, width 80; set: height 24,
width 80
```

```
00:34:04: SSH2 1: input: packet len 96
00:34:04: SSH2 1: partial packet 16, need 80, maclen 16
00:34:04: SSH2 1: MAC #9 ok
00:34:04: SSH2 1: input: padlen 11
00:34:04: SSH2 1: received packet type 98
00:34:04: SSH2 1: x11-req request
00:34:04: SSH2 1: ssh_receive: 48 bytes received
00:34:04: SSH2 1: input: packet len 32
00:34:04: SSH2 1: partial packet 16, need 16, maclen 16
00:34:04: SSH2 1: MAC #10 ok
00:34:04: SSH2 1: input: padlen 12
00:34:04: SSH2 1: received packet type 98
00:34:04: SSH2 1: shell request
00:34:04: SSH2 1: shell message received
00:34:04: SSH2 1: starting shell for vty
00:34:04: SSH2 1: send: len 48 (includes padlen 18)
00:34:04: SSH2 1: done calc MAC out #7
00:34:07: SSH2 1: ssh_receive: 48 bytes received
00:34:07: SSH2 1: input: packet len 32
00:34:07: SSH2 1: partial packet 16, need 16, maclen 16
00:34:07: SSH2 1: MAC #11 ok
00:34:07: SSH2 1: input: padlen 17
00:34:07: SSH2 1: received packet type 94
00:34:07: SSH2 1: send: len 32 (includes padlen 17)
00:34:07: SSH2 1: done calc MAC out #8
00:34:07: SSH2 1: ssh_receive: 48 bytes received
00:34:07: SSH2 1: input: packet len 32
00:34:07: SSH2 1: partial packet 16, need 16, maclen 16
00:34:07: SSH2 1: MAC #12 ok
00:34:07: SSH2 1: input: padlen 17
00:34:07: SSH2 1: received packet type 94
00:34:07: SSH2 1: send: len 32 (includes padlen 17)
00:34:07: SSH2 1: done calc MAC out #9
00:34:07: SSH2 1: ssh_receive: 48 bytes received
00:34:07: SSH2 1: input: packet len 32
00:34:07: SSH2 1: partial packet 16, need 16, maclen 16
00:34:07: SSH2 1: MAC #13 ok
00:34:07: SSH2 1: input: padlen 17
00:34:07: SSH2 1: received packet type 94
00:34:07: SSH2 1: send: len 32 (includes padlen 17)
00:34:07: SSH2 1: done calc MAC out #10
00:34:08: SSH2 1: ssh_receive: 48 bytes received
00:34:08: SSH2 1: input: packet len 32
00:34:08: SSH2 1: partial packet 16, need 16, maclen 16
00:34:08: SSH2 1: MAC #14 ok
00:34:08: SSH2 1: input: padlen 17
00:34:08: SSH2 1: received packet type 94
00:34:08: SSH2 1: send: len 32 (includes padlen 17)
00:34:08: SSH2 1: done calc MAC out #11
00:34:08: SSH2 1: ssh_receive: 48 bytes received
00:34:08: SSH2 1: input: packet len 32
00:34:08: SSH2 1: partial packet 16, need 16, maclen 16
00:34:08: SSH2 1: MAC #15 ok
00:34:08: SSH2 1: input: padlen 17
00:34:08: SSH2 1: received packet type 94
00:34:08: SSH2 1: send: len 32 (includes padlen 16)
00:34:08: SSH2 1: done calc MAC out #12
00:34:08: SSH2 1: send: len 48 (includes padlen 18)
00:34:08: SSH2 1: done calc MAC out #13
00:34:08: SSH2 1: send: len 16 (includes padlen 6)
00:34:08: SSH2 1: done calc MAC out #14
00:34:08: SSH2 1: send: len 16 (includes padlen 6)
00:34:08: SSH2 1: done calc MAC out #15
00:34:08: SSH1: Session terminated normally
```


Configuration Examples for Secure Shell Version 2 Support

Example: Configuring Secure Shell Version 2

```
Device# configure terminal
Device(config)# ip ssh version 2
```

Example: Starting an Encrypted Session with a Remote Device

```
Device# ssh -v 2 -c aes256-cbc -m hmac-sha1-160 -l shaship 10.76.82.24
```

Example: Configuring Server-Side SCP

The following example shows how to configure the server-side functionality for SCP. This example also configures AAA authentication and authorization on the device. This example uses a locally defined username and password.

```
Device# configure terminal
Device(config)# aaa new-model
Device(config)# aaa authentication login default local
Device(config)# aaa authorization exec default local
Device(config)# username samplename privilege 15 password password1
Device(config)# ip ssh time-out 120
Device(config)# ip ssh authentication-retries 3
Device(config)# ip scp server enable
```

Example: Setting an SNMP Trap

The following example shows that an SNMP trap is set. The trap notification is generated automatically when the SSH session terminates. In the example, a.b.c.d is the IP address of the SSH client.

```
snmp-server
snmp-server host a.b.c.d public tty
```

The following is sample output from the **debug snmp packet** command. The output provides SNMP trap information for an SSH session.

```
Device1# debug snmp packet

SNMP packet debugging is on
Device1# ssh -l lab 10.0.0.2
Password:

Device2# exit

[Connection to 10.0.0.2 closed by foreign host]
Device1#
*Jul 18 10:18:42.619: SNMP: Queuing packet to 10.0.0.2
```

```
*Jul 18 10:18:42.619: SNMP: V1 Trap, ent cisco, addr 10.0.0.1, gentrap 6, spectrap 1
local.9.3.1.1.2.1 = 6
tcpConnEntry.1.10.0.0.1.22.10.0.0.2.55246 = 4
ltcpConnEntry.5.10.0.0.1.22.10.0.0.2.55246 = 1015
ltcpConnEntry.1.10.0.0.1.22.10.0.0.2.55246 = 1056
ltcpConnEntry.2.10.0.0.1.22.10.0.0.2.55246 = 1392
local.9.2.1.18.2 = lab
*Jul 18 10:18:42.879: SNMP: Packet sent via UDP to 10.0.0.2

Device1#
```

Examples: SSH Keyboard Interactive Authentication

Example: Enabling Client-Side Debugs

The following example shows that the client-side debugs are turned on, and the maximum number of prompts is six (three for the SSH keyboard interactive authentication method and three for the password authentication method).

```
Password:
Password:
Password:
Password:
Password:
Password: cisco123
Last login: Tue Dec 6 13:15:21 2005 from 10.76.248.213
user1@courier:~> exit
logout
[Connection to 10.76.248.200 closed by foreign host]
Device1# debug ip ssh client

SSH Client debugging is on

Device1# ssh -l lab 10.1.1.3

Password:
*Nov 17 12:50:53.199: SSH0: sent protocol version id SSH-1.99-Cisco-1.25
*Nov 17 12:50:53.199: SSH CLIENT0: protocol version id is - SSH-1.99-Cisco-1.25
*Nov 17 12:50:53.199: SSH CLIENT0: sent protocol version id SSH-1.99-Cisco-1.25
*Nov 17 12:50:53.199: SSH CLIENT0: protocol version exchange successful
*Nov 17 12:50:53.203: SSH0: protocol version id is - SSH-1.99-Cisco-1.25
*Nov 17 12:50:53.335: SSH CLIENT0: key exchange successful and encryption on
*Nov 17 12:50:53.335: SSH2 CLIENT 0: using method keyboard-interactive
Password:
Password:
Password:
*Nov 17 12:51:01.887: SSH2 CLIENT 0: using method password authentication
Password:
Password: lab
Device2>

*Nov 17 12:51:11.407: SSH2 CLIENT 0: SSH2_MSG_USERAUTH_SUCCESS message received
*Nov 17 12:51:11.407: SSH CLIENT0: user authenticated
*Nov 17 12:51:11.407: SSH2 CLIENT 0: pty-req request sent
*Nov 17 12:51:11.411: SSH2 CLIENT 0: shell request sent
*Nov 17 12:51:11.411: SSH CLIENT0: session open
```

Example: Enabling ChPass with a Blank Password Change

In the following example, the ChPass feature is enabled, and a blank password change is accomplished using the SSH Keyboard Interactive Authentication method. A TACACS+ access control server (ACS) is used as the back-end AAA server.

```
Device1# ssh -l cisco 10.1.1.3

Password:
Old Password: cisco
New Password: cisco123
Re-enter New password: cisco123

Device2> exit

[Connection to 10.1.1.3 closed by foreign host]
```

Example: Enabling ChPass and Changing the Password on First Login

In the following example, the ChPass feature is enabled and TACACS+ ACS is used as the back-end server. The password is changed on the first login using the SSH keyboard interactive authentication method.

```
Device1# ssh -l cisco 10.1.1.3

Password: cisco
Your password has expired.
Enter a new one now.
New Password: cisco123
Re-enter New password: cisco123

Device2> exit

[Connection to 10.1.1.3 closed by foreign host]

Device1# ssh -l cisco 10.1.1.3

Password:cisco1
Your password has expired.
Enter a new one now.
New Password: cisco
Re-enter New password: cisco12
The New and Re-entered passwords have to be the same.
Try again.
New Password: cisco
Re-enter New password: cisco

Device2>
```

Example: Enabling ChPass and Expiring the Password After Three Logins

In the following example, the ChPass feature is enabled and TACACS+ ACS is used as the back-end AAA server. The password expires after three logins using the SSH keyboard interactive authentication method.

```
Device# ssh -l cisco 10.1.1.3

Password: cisco

Device2> exit
```

```

[Connection to 10.1.1.3 closed by foreign host]

Device1# ssh -l cisco 10.1.1.3

Password: cisco

Device2> exit

Device1# ssh -l cisco 10.1.1.3

Password: cisco

Device2> exit

[Connection to 10.1.1.3 closed by foreign host]

Device1# ssh -l cisco 10.1.1.3

Password: cisco
Your password has expired.
Enter a new one now.
New Password: cisco123
Re-enter New password: cisco123

Device2>

```

Example: SNMP Debugging

The following is sample output from the **debug snmp packet** command. The output provides SNMP trap information for an SSH session.

```

Device1# debug snmp packet

SNMP packet debugging is on
Device1# ssh -l lab 10.0.0.2
Password:

Device2# exit

[Connection to 10.0.0.2 closed by foreign host]
Device1#
*Jul 18 10:18:42.619: SNMP: Queuing packet to 10.0.0.2
*Jul 18 10:18:42.619: SNMP: V1 Trap, ent cisco, addr 10.0.0.1, gentrap 6, spectrap 1
local.9.3.1.1.2.1 = 6
tcpConnEntry.1.10.0.0.1.22.10.0.0.2.55246 = 4
ltcpConnEntry.5.10.0.0.1.22.10.0.0.2.55246 = 1015
ltcpConnEntry.1.10.0.0.1.22.10.0.0.2.55246 = 1056
ltcpConnEntry.2.10.0.0.1.22.10.0.0.2.55246 = 1392
local.9.2.1.18.2 = lab
*Jul 18 10:18:42.879: SNMP: Packet sent via UDP to 10.0.0.2

Device1#

```

Examples: SSH Debugging Enhancements

The following is sample output from the **debug ip ssh detail** command. The output provides debugging information about the SSH protocol and channel requests.

```

Device# debug ip ssh detail

```

```

00:04:22: SSH0: starting SSH control process
00:04:22: SSH0: sent protocol version id SSH-1.99-Cisco-1.25
00:04:22: SSH0: protocol version id is - SSH-1.99-Cisco-1.25
00:04:22: SSH2 0: SSH2_MSG_KEXINIT sent
00:04:22: SSH2 0: SSH2_MSG_KEXINIT received
00:04:22: SSH2:kex: client->server enc:aes128-cbc mac:hmac-shal
00:04:22: SSH2:kex: server->client enc:aes128-cbc mac:hmac-shal
00:04:22: SSH2 0: expecting SSH2_MSG_KEXDH_INIT
00:04:22: SSH2 0: SSH2_MSG_KEXDH_INIT received
00:04:22: SSH2: kex_derive_keys complete
00:04:22: SSH2 0: SSH2_MSG_NEWKEYS sent
00:04:22: SSH2 0: waiting for SSH2_MSG_NEWKEYS
00:04:22: SSH2 0: SSH2_MSG_NEWKEYS received
00:04:24: SSH2 0: authentication successful for lab
00:04:24: SSH2 0: channel open request
00:04:24: SSH2 0: pty-req request
00:04:24: SSH2 0: setting TTY - requested: height 24, width 80; set: height 24, width 80
00:04:24: SSH2 0: shell request
00:04:24: SSH2 0: shell message received
00:04:24: SSH2 0: starting shell for vty
00:04:38: SSH0: Session terminated normally

```

The following is sample output from the **debug ip ssh packet** command. The output provides debugging information about the SSH packet.

```
Device# debug ip ssh packet
```

```

00:05:43: SSH2 0: send:packet of length 280 (length also includes padlen of 4)
00:05:43: SSH2 0: ssh_receive: 64 bytes received
00:05:43: SSH2 0: input: total packet length of 280 bytes
00:05:43: SSH2 0: partial packet length(block size)8 bytes,needed 272 bytes, maclen 0
00:05:43: SSH2 0: ssh_receive: 64 bytes received
00:05:43: SSH2 0: partial packet length(block size)8 bytes,needed 272 bytes, maclen 0
00:05:43: SSH2 0: ssh_receive: 64 bytes received
00:05:43: SSH2 0: partial packet length(block size)8 bytes,needed 272 bytes, maclen 0
00:05:43: SSH2 0: ssh_receive: 64 bytes received
00:05:43: SSH2 0: partial packet length(block size)8 bytes,needed 272 bytes, maclen 0
00:05:43: SSH2 0: ssh_receive: 24 bytes received
00:05:43: SSH2 0: partial packet length(block size)8 bytes,needed 272 bytes, maclen 0
00:05:43: SSH2 0: input: padlength 4 bytes
00:05:43: SSH2 0: ssh_receive: 64 bytes received
00:05:43: SSH2 0: input: total packet length of 144 bytes
00:05:43: SSH2 0: partial packet length(block size)8 bytes,needed 136 bytes, maclen 0
00:05:43: SSH2 0: ssh_receive: 64 bytes received
00:05:43: SSH2 0: partial packet length(block size)8 bytes,needed 136 bytes, maclen 0
00:05:43: SSH2 0: ssh_receive: 16 bytes received
00:05:43: SSH2 0: partial packet length(block size)8 bytes,needed 136 bytes, maclen 0
00:05:43: SSH2 0: input: padlength 6 bytes
00:05:43: SSH2 0: signature length 143
00:05:43: SSH2 0: send:packet of length 448 (length also includes padlen of 7)
00:05:43: SSH2 0: send:packet of length 16 (length also includes padlen of 10)
00:05:43: SSH2 0: newkeys: mode 1
00:05:43: SSH2 0: ssh_receive: 16 bytes received
00:05:43: SSH2 0: input: total packet length of 16 bytes
00:05:43: SSH2 0: partial packet length(block size)8 bytes,needed 8 bytes, maclen 0
00:05:43: SSH2 0: input: padlength 10 bytes
00:05:43: SSH2 0: newkeys: mode 0
00:05:43: SSH2 0: ssh_receive: 52 bytes received
00:05:43: SSH2 0: input: total packet length of 32 bytes
00:05:43: SSH2 0: partial packet length(block size)16 bytes,needed 16 bytes, maclen 20
00:05:43: SSH2 0: MAC compared for #3 :ok

```

Additional References for Secure Shell Version 2 Support

Standards

Standards	Title
IETF Secure Shell Version 2 Draft Standards	Internet Engineering Task Force website

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

Feature Information for Secure Shell Version 2 Support

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Table 129: Feature Information for Secure Shell Version 2 Support

Feature Name	Releases	Feature Information
Secure Shell Version 2 Client and Server Support	Cisco IOS XE Release 3.4SG	The Cisco image was updated to provide for the automatic generation of SNMP traps when an SSH session terminates. This feature was supported on CAT2960, CAT3560E, CAT3560X, CAT3750, CAT3750E, CAT3750X, CAT4500.

Feature Name	Releases	Feature Information
Secure Shell Version 2 Enhancements	Cisco IOS XE Release 3.4SG	<p>The Secure Shell Version 2 Enhancements feature includes a number of additional capabilities such as support for VRF-Aware SSH, SSH debug enhancements, and DH Group 14 and Group 16 exchange support.</p> <p>This feature was supported on CAT2960, CAT3560E, CAT3560X, CAT3750, CAT3750E, CAT3750X, CAT4500.</p> <p>Note The VRF-Aware SSH feature is supported depending on your release.</p> <p>The following commands were introduced or modified: debug ip ssh, and ip ssh dh min size.</p>
Secure Shell Version 2 Enhancements for RSA Keys	Cisco IOS XE Release 3.4SG	<p>The Secure Shell Version 2 Enhancements for RSA Keys feature includes a number of additional capabilities to support RSA key-based user authentication for SSH and SSH server host key storage and verification.</p> <p>This feature was supported on CAT2960, CAT3560E, CAT3560X, CAT3750, CAT3750E, CAT3750X, CAT4500.</p>
Secure Shell Version 2 Support	Cisco IOS XE Release 3.4SG	<p>The Secure Shell Version 2 Support feature allows you to configure Secure Shell (SSH) Version 2 (SSH Version 1 support was implemented in an earlier Cisco software release). SSH runs on top of a reliable transport layer and provides strong authentication and encryption capabilities.</p> <p>This feature was supported on CAT2960, CAT3560E, CAT3560X, CAT3750, CAT3750E, CAT3750X, CAT4500.</p> <p>The following commands were introduced or modified: debug ip ssh, ip ssh min dh size, ip ssh rsa keypair-name, ip ssh version, and ssh.</p>

Feature Name	Releases	Feature Information
SSH Keyboard Interactive Authentication	Cisco IOS XE Release 3.4SG	<p>The SSH Keyboard Interactive Authentication feature, also known as Generic Message Authentication for SSH, is a method that can be used to implement different types of authentication mechanisms. Basically, any currently supported authentication method that requires only user input can be performed with this feature.</p> <p>This feature was supported on CAT2960, CAT3560E, CAT3560X, CAT3750, CAT3750E, CAT3750X, CAT4500.</p>



CHAPTER 64

X.509v3 Certificates for SSH Authentication

The X.509v3 Certificates for SSH Authentication feature uses public key algorithm (PKI) for server and user authentication, and allows the Secure Shell (SSH) protocol to verify the identity of the owner of a key pair via digital certificates, signed and issued by a Certificate Authority (CA).

This module describes how to configure server and user certificate profiles for a digital certificate.

- [Prerequisites for X.509v3 Certificates for SSH Authentication, on page 1271](#)
- [Restrictions for X.509v3 Certificates for SSH Authentication, on page 1271](#)
- [Information About X.509v3 Certificates for SSH Authentication, on page 1272](#)
- [How to Configure X.509v3 Certificates for SSH Authentication, on page 1273](#)
- [Verifying the Server and User Authentication Using Digital Certificates , on page 1276](#)
- [Configuration Examples for X.509v3 Certificates for SSH Authentication, on page 1280](#)
- [Additional References for X.509v3 Certificates for SSH Authentication, on page 1281](#)
- [Feature Information for X.509v3 Certificates for SSH Authentication, on page 1281](#)

Prerequisites for X.509v3 Certificates for SSH Authentication

The X.509v3 Certificates for SSH Authentication feature replaces the **ip ssh server authenticate user** command with the **ip ssh server algorithm authentication** command. Configure the **default ip ssh server authenticate user** command to remove the **ip ssh server authenticate user** command from the configuration. The IOS secure shell (SSH) server will start using the **ip ssh server algorithm authentication** command.

When you configure the **ip ssh server authenticate user** command, the following message is displayed:



Warning

SSH command accepted; but this CLI will be deprecated soon. Please move to new CLI **ip ssh server algorithm authentication**. Please configure the “**default ip ssh server authenticate user**” to make the CLI ineffective.

Restrictions for X.509v3 Certificates for SSH Authentication

- The X.509v3 Certificates for SSH Authentication feature implementation is applicable only on the Cisco IOS Secure Shell (SSH) server side.

- The Cisco IOS SSH server supports only the x509v3-ssh-rsa algorithm-based certificate for server and user authentication.

Information About X.509v3 Certificates for SSH Authentication

X.509v3 Certificates for SSH Authentication Overview

The Secure Shell (SSH) protocol provides a secure remote access connection to network devices. The communication between the client and server is encrypted.

There are two SSH protocols that use public key cryptography for authentication. The Transport Layer Protocol, uses a digital signature algorithm (called the public key algorithm) to authenticate the server to the client. And the User Authentication Protocol uses a digital signature to authenticate (public key authentication) the client to the server.

The validity of the authentication depends upon the strength of the linkage between the public signing key and the identity of the signer. Digital certificates, such as those in X.509 Version 3 (X.509v3), are used to provide identity management. X.509v3 uses a chain of signatures by a trusted root certification authority and intermediate certificate authorities to bind a public signing key to a specific digital identity. This implementation allows the use of a public key algorithm for server and user authentication, and allows SSH to verify the identity of the owner of a key pair via digital certificates, signed and issued by a Certificate Authority (CA).

Server and User Authentication Using X.509v3

For server authentication, the Secure shell (SSH) server sends its own certificate to the SSH client for verification. This server certificate is associated with the trustpoint configured in the server certificate profile (ssh-server-cert-profile-server configuration mode).

For user authentication, the SSH client sends the user's certificate to the IOS SSH server for verification. The SSH server validates the incoming user certificate using public key infrastructure (PKI) trustpoints configured in the server certificate profile (ssh-server-cert-profile-user configuration mode).

By default, certificate-based authentication is enabled for server and user at the IOS SSH server end.

OCSP Response Stapling

The Online Certificate Status Protocol (OCSP) enables applications to determine the (revocation) state of an identified certificate. This protocol specifies the data that needs to be exchanged between an application checking the status of a certificate and the server providing that status. An OCSP client issues a status request to an OCSP responder and suspends acceptance of the certificate until a response is received. An OCSP response at a minimum consists of a responseStatus field that indicates the processing status of the a request.

For the public key algorithms, the key format consists of a sequence of one or more X.509v3 certificates followed by a sequence of zero or more OCSP responses.

The X.509v3 Certificate for SSH Authentication feature uses OCSP Response Stapling. By using OCSP response stapling, a device obtains the revocation information of its own certificate by contacting the OCSP server and then stapling the result along with its certificates and sending the information to the peer rather than having the peer contact the OCSP responder.

How to Configure X.509v3 Certificates for SSH Authentication

Configuring Digital Certificates for Server Authentication

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip ssh server algorithm hostkey {x509v3-ssh-rsa [ssh-rsa] | ssh-rsa [x509v3-ssh-rsa]}**
4. **ip ssh server certificate profile**
5. **server**
6. **trustpoint sign *PKI-trustpoint-name***
7. **ocsp-response include**
8. **end**
9. **line vty line_number [ending_line_number]**
10. **transport input ssh**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Switch> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 3	ip ssh server algorithm hostkey {x509v3-ssh-rsa [ssh-rsa] ssh-rsa [x509v3-ssh-rsa]} Example: <pre>Switch(config)# ip ssh server algorithm hostkey x509v3-ssh-rsa</pre>	Defines the order of host key algorithms. Only the configured algorithm is negotiated with the Secure Shell (SSH) client. <p>Note The IOS SSH server must have at least one configured host key algorithm:</p> <ul style="list-style-type: none"> • x509v3-ssh-rsa—certificate-based authentication • ssh-rsa—public key-based authentication
Step 4	ip ssh server certificate profile Example: <pre>Switch(config)# ip ssh server certificate profile</pre>	Configures server and user certificate profiles and enters SSH certificate profile configuration mode.

	Command or Action	Purpose
Step 5	server Example: <pre>Switch(ssh-server-cert-profile)# server</pre>	Configures server certificate profile and enters SSH server certificate profile server configuration mode. <ul style="list-style-type: none"> The server profile is used to send out the certificate of the server to the SSH client during server authentication.
Step 6	trustpoint sign <i>PKI-trustpoint-name</i> Example: <pre>Switch(ssh-server-cert-profile-server)# trustpoint sign trust1</pre>	Attaches the public key infrastructure (PKI) trustpoint to the server certificate profile. <ul style="list-style-type: none"> The SSH server uses the certificate associated with this PKI trustpoint for server authentication.
Step 7	ocsp-response include Example: <pre>Switch(ssh-server-cert-profile-server)# ocsp-response include</pre>	(Optional) Sends the Online Certificate Status Protocol (OCSP) response or OCSP stapling along with the server certificate. <p>Note By default, no OCSP response is sent along with the server certificate.</p>
Step 8	end Example: <pre>Switch(ssh-server-cert-profile-server)# end</pre>	Exits SSH server certificate profile server configuration mode and returns to privileged EXEC mode.
Step 9	line vty line_number [ending_line_number] Example: <pre>Switch(config)# line vty line_number [ending_line_number]</pre>	Enters line configuration mode to configure the virtual terminal line settings. For line_number and ending_line_number, specify a pair of lines. The range is 0 to 15.
Step 10	transport input ssh Example: <pre>Switch(config-line)#transport input ssh</pre>	Specifies that the Switch prevent non-SSH Telnet connections. This limits the router to only SSH connections.

Configuring Digital Certificates for User Authentication

SUMMARY STEPS

- enable
- configure terminal
- ip ssh server algorithm authentication {publickey | keyboard | password}
- ip ssh server algorithm publickey {x509v3-ssh-rsa [ssh-rsa] | ssh-rsa [x509v3-ssh-rsa]}
- ip ssh server certificate profile
- user
- trustpoint verify *PKI-trustpoint-name*
- ocsp-response required

9. **end**
10. **line vty line_number** [*ending_line_number*]
11. **transport input ssh**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Switch> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 3	ip ssh server algorithm authentication {publickey keyboard password} Example: <pre>Switch(config)# ip ssh server algorithm authentication publickey</pre>	Defines the order of user authentication algorithms. Only the configured algorithm is negotiated with the Secure Shell (SSH) client. <p>Note</p> <ul style="list-style-type: none"> • The IOS SSH server must have at least one configured user authentication algorithm. • To use the certificate method for user authentication, the publickey keyword must be configured.
Step 4	ip ssh server algorithm publickey {x509v3-ssh-rsa [ssh-rsa] ssh-rsa [x509v3-ssh-rsa]} Example: <pre>Switch(config)# ip ssh server algorithm publickey x509v3-ssh-rsa</pre>	Defines the order of public key algorithms. Only the configured algorithm is accepted by the SSH client for user authentication. <p>Note</p> The IOS SSH client must have at least one configured public key algorithm: <ul style="list-style-type: none"> • x509v3-ssh-rsa—Certificate-based authentication • ssh-rsa—Public-key-based authentication
Step 5	ip ssh server certificate profile Example: <pre>Switch(config)# ip ssh server certificate profile</pre>	Configures server certificate profile and user certificate profile and enters SSH certificate profile configuration mode.
Step 6	user Example: <pre>Switch(ssh-server-cert-profile)# user</pre>	Configures user certificate profile and enters SSH server certificate profile user configuration mode.
Step 7	trustpoint verify PKI-trustpoint-name Example:	Configures the public key infrastructure (PKI) trustpoint that is used to verify the incoming user certificate.

	Command or Action	Purpose
	Switch(ssh-server-cert-profile-user)# trustpoint verify trust2	Note Configure multiple trustpoints by executing the same command multiple times. A maximum of 10 trustpoints can be configured.
Step 8	ocsp-response required Example: Switch(ssh-server-cert-profile-user)# ocsp-response required	(Optional) Mandates the presence of the Online Certificate Status Protocol (OCSP) response with the incoming user certificate. Note By default, the user certificate is accepted without an OCSP response.
Step 9	end Example: Switch(ssh-server-cert-profile-user)# end	Exits SSH server certificate profile user configuration mode and returns to privileged EXEC mode.
Step 10	line vty line_number [ending_line_number] Example: Switch(config)# line vty line_number [ending_line_number]	Enters line configuration mode to configure the virtual terminal line settings. For line_number and ending_line_number, specify a pair of lines. The range is 0 to 15.
Step 11	transport input ssh Example: Switch(config-line)#transport input ssh	Specifies that the Switch prevent non-SSH Telnet connections. This limits the router to only SSH connections.

Verifying the Server and User Authentication Using Digital Certificates

SUMMARY STEPS

1. enable
2. show ip ssh
3. debug ip ssh detail
4. show log
5. debug ip packet
6. show log

DETAILED STEPS

Step 1 enable

Enables privileged EXEC mode.

- Enter your password if prompted.

Example:

```
Device> enable
```

Step 2 show ip ssh

Displays the currently configured authentication methods. To confirm the use of certificate-based authentication, ensure that the x509v3-ssh-rsa algorithm is the configured host key algorithm.

Example:

```
Device# show ip ssh

SSH Enabled - version 1.99
Authentication methods:publickey,keyboard-interactive,password
Authentication Publickey Algorithms:x509v3-ssh-rsa,ssh-rsa
Hostkey Algorithms:x509v3-ssh-rsa,ssh-rsa
Authentication timeout: 120 secs; Authentication retries: 3
Minimum expected Diffie Hellman key size : 1024 bits
```

Step 3 debug ip ssh detail

Turns on debugging messages for SSH details.

Example:

```
Device# debug ip ssh detail

ssh detail messages debugging is on
```

Step 4 show log

Shows the debug message log.

Example:

```
Device# show log

Syslog logging: enabled (0 messages dropped, 9 messages rate-limited, 0 flushes, 0 overruns, xml
disabled, filtering disabled)

No Active Message Discriminator.

No Inactive Message Discriminator.

Console logging: disabled
Monitor logging: level debugging, 0 messages logged, xml disabled,
filtering disabled
Buffer logging: level debugging, 233 messages logged, xml disabled,
filtering disabled
Exception Logging: size (4096 bytes)
Count and timestamp logging messages: disabled
File logging: disabled
Persistent logging: disabled

No active filter modules.

Trap logging: level informational, 174 message lines logged
Logging Source-Interface: VRF Name:
```

```

Log Buffer (4096 bytes):
5 IST: SSH2 CLIENT 0: SSH2_MSG_KEXINIT sent
*Sep 6 14:44:08.496 IST: SSH0: protocol version id is - SSH-1.99-Cisco-1.25
*Sep 6 14:44:08.496 IST: SSH2 0: kexinit sent: kex algo =
diffie-hellman-group-exchange-sha1,diffie-hellman-group14-sha1
*Sep 6 14:44:08.496 IST: SSH2 0: Server certificate trustpoint not found. Skipping hostkey algo =
x509v3-ssh-rsa
*Sep 6 14:44:08.496 IST: SSH2 0: kexinit sent: hostkey algo = ssh-rsa
*Sep 6 14:44:08.496 IST: SSH2 0: kexinit sent: encryption algo = aes128-ctr,aes192-ctr,aes256-ctr
*Sep 6 14:44:08.496 IST: SSH2 0: kexinit sent: mac algo =
hmac-sha2-256,hmac-sha2-512,hmac-sha1,hmac-sha1-96
*Sep 6 14:44:08.496 IST: SSH2 0: SSH2_MSG_KEXINIT sent
*Sep 6 14:44:08.496 IST: SSH2 0: SSH2_MSG_KEXINIT received
*Sep 6 14:44:08.496 IST: SSH2 0: kex: client->server enc:aes128-ctr mac:hmac-sha2-256
*Sep 6 14:44:08.496 IST: SSH2 0: kex: server->client enc:aes128-ctr mac:hmac-sha2-256
*Sep 6 14:44:08.496 IST: SSH2 0: Using hostkey algo = ssh-rsa
*Sep 6 14:44:08.496 IST: SSH2 0: Using kex_algo = diffie-hellman-group-exchange-sha1
*Sep 6 14:44:08.497 IST: SSH2 CLIENT 0: SSH2_MSG_KEXINIT received
*Sep 6 14:44:08.497 IST: SSH2 CLIENT 0: kex: server->client enc:aes128-ctr mac:hmac-sha2-256
*Sep 6 14:44:08.497 IST: SSH2 CLIENT 0: kex: client->server enc:aes128-ctr mac:hmac-sha2-256
*Sep 6 14:44:08.497 IST: SSH2 CLIENT 0: Using hostkey algo = ssh-rsa
*Sep 6 14:44:08.497 IST: SSH2 CLIENT 0: Using kex_algo = diffie-hellman-group-exchange-sha1
*Sep 6 14:44:08.497 IST: SSH2 CLIENT 0: SSH2_MSG_KEX_DH_GEX_REQUEST sent
*Sep 6 14:44:08.497 IST: SSH2 CLIENT 0: Range sent- 2048 < 2048 < 4096
*Sep 6 14:44:08.497 IST: SSH2 0: SSH2_MSG_KEX_DH_GEX_REQUEST received
*Sep 6 14:44:08.497 IST: SSH2 0: Range sent by client is - 2048 < 2048 < 4096
*Sep 6 14:44:08.497 IST: SSH2 0: Modulus size established : 2048 bits
*Sep 6 14:44:08.510 IST: SSH2 0: expecting SSH2_MSG_KEX_DH_GEX_INIT
*Sep 6 14:44:08.510 IST: SSH2 CLIENT 0: SSH2_MSG_KEX_DH_GEX_GROUP received
*Sep 6 14:44:08.510 IST: SSH2 CLIENT 0: Server has chosen 2048 -bit dh keys
*Sep 6 14:44:08.523 IST: SSH2 CLIENT 0: expecting SSH2_MSG_KEX_DH_GEX_REPLY
*Sep 6 14:44:08.524 IST: SSH2 0: SSH2_MSG_KEXDH_INIT received
*Sep 6 14:44:08.555 IST: SSH2: kex_derive_keys complete
*Sep 6 14:44:08.555 IST: SSH2 0: SSH2_MSG_NEWKEYS sent
*Sep 6 14:44:08.555 IST: SSH2 0: waiting for SSH2_MSG_NEWKEYS
*Sep 6 14:44:08.555 IST: SSH2 CLIENT 0: SSH2_MSG_KEX_DH_GEX_REPLY received
*Sep 6 14:44:08.555 IST: SSH2 CLIENT 0: Skipping ServerHostKey Validation
*Sep 6 14:44:08.571 IST: SSH2 CLIENT 0: signature length 271
*Sep 6 14:44:08.571 IST: SSH2: kex_derive_keys complete
*Sep 6 14:44:08.571 IST: SSH2 CLIENT 0: SSH2_MSG_NEWKEYS sent
*Sep 6 14:44:08.571 IST: SSH2 CLIENT 0: waiting for SSH2_MSG_NEWKEYS
*Sep 6 14:44:08.571 IST: SSH2 CLIENT 0: SSH2_MSG_NEWKEYS received
*Sep 6 14:44:08.571 IST: SSH2 0: SSH2_MSG_NEWKEYS received
*Sep 6 14:44:08.571 IST: SSH2 0: Authentications that can continue =
publickey,keyboard-interactive,password
*Sep 6 14:44:08.572 IST: SSH2 0: Using method = none
*Sep 6 14:44:08.572 IST: SSH2 0: Authentications that can continue =
publickey,keyboard-interactive,password
*Sep 6 14:44:08.572 IST: SSH2 0: Using method = keyboard-interactive
*Sep 6 14:44:11.983 IST: SSH2 0: authentication successful for cisco
*Sep 6 14:44:11.984 IST: %SEC_LOGIN-5-LOGIN_SUCCESS: Login Success [user: cisco] [Source:
192.168.121.40] [localport: 22] at 14:44:11 IST Thu Sep 6 2018
*Sep 6 14:44:11.984 IST: SSH2 0: channel open request
*Sep 6 14:44:11.985 IST: SSH2 0: pty-req request
*Sep 6 14:44:11.985 IST: SSH2 0: setting TTY - requested: height 24, width 80; set: height 24, width
80
*Sep 6 14:44:11.985 IST: SSH2 0: shell request
*Sep 6 14:44:11.985 IST: SSH2 0: shell message received
*Sep 6 14:44:11.985 IST: SSH2 0: starting shell for vty
*Sep 6 14:44:22.066 IST: %SYS-6-LOGOUT: User cisco has exited tty session 1(192.168.121.40)
*Sep 6 14:44:22.166 IST: SSH0: Session terminated normally
*Sep 6 14:44:22.167 IST: SSH CLIENT0: Session terminated normally

```


Step 5 **debug ip packet**

Turns on debugging for IP packet details.

Example:

```
Device# debug ip packet
```

Step 6 **show log**

Shows the debug message log.

Example:

```
Device# show log
```

```
yslog logging: enabled (0 messages dropped, 9 messages rate-limited, 0 flushes, 0 overruns, xml
disabled, filtering disabled)
```

```
No Active Message Discriminator.
```

```
No Inactive Message Discriminator.
```

```
Console logging: disabled
Monitor logging: level debugging, 0 messages logged, xml disabled,
                  filtering disabled
Buffer logging:  level debugging, 1363 messages logged, xml disabled,
                  filtering disabled
Exception Logging: size (4096 bytes)
Count and timestamp logging messages: disabled
File logging: disabled
Persistent logging: disabled
```

```
No active filter modules.
```

```
Trap logging: level informational, 176 message lines logged
Logging Source-Interface:      VRF Name:
```

```
Log Buffer (4096 bytes):
bleid=0, s=192.168.121.40 (local), d=192.168.121.40 (FortyGigabitEthernet1/0/1), routed via RIB
*Sep  6 14:45:45.177 IST: IP: s=192.168.121.40 (local), d=192.168.121.40 (FortyGigabitEthernet1/0/1),
  len 40, sending
*Sep  6 14:45:45.177 IST: IP: s=192.168.121.40 (local), d=192.168.121.40 (FortyGigabitEthernet1/0/1),
  len 40, output feature, NAT Inside(8), rtype 1, forus FALSE, sendself FALSE, mtu 0, fwdchk FALSE
*Sep  6 14:45:45.177 IST: IP: tableid=0, s=192.168.121.40 (FortyGigabitEthernet1/0/1), d=192.168.121.40
  (FortyGigabitEthernet1/0/1), routed via RIB
*Sep  6 14:45:45.177 IST: IP: tableid=0, s=192.168.121.40 (FortyGigabitEthernet1/0/1), d=192.168.121.40
  (FortyGigabitEthernet1/0/1), routed via RIB
*Sep  6 14:45:45.177 IST: IP: s=192.168.121.40 (local), d=192.168.121.40, len 40, local feature,
  feature skipped, NAT(2), rtype 0, forus FALSE, sendself FALSE, mtu 0, fwdchk FALSE
*Sep  6 14:45:45.178 IST: IP: tableid=0, s=192.168.121.40 (local), d=192.168.121.40
  (FortyGigabitEthernet1/0/1), routed via RIB
*Sep  6 14:45:45.178 IST: IP: s=192.168.121.40 (local), d=192.168.121.40 (FortyGigabitEthernet1/0/1),
  len 40, sending
*Sep  6 14:45:45.178 IST: IP: s=192.168.121.40 (local), d=192.168.121.40 (FortyGigabitEthernet1/0/1),
  len 40, output feature, NAT Inside(8), rtype 1, forus FALSE, sendself FALSE, mtu 0, fwdchk FALSE
*Sep  6 14:45:45.178 IST: IP: tableid=0, s=192.168.121.40 (FortyGigabitEthernet1/0/1), d=192.168.121.40
  (FortyGigabitEthernet1/0/1), routed via RIB
*Sep  6 14:45:45.178 IST: IP: s=192.168.121.40 (local), d=192.168.121.40, len 40, local feature,
  feature skipped, NAT(2), rtype 0, forus FALSE, sendself FALSE, mtu 0, fwdchk FALSE
*Sep  6 14:45:45.178 IST: IP: tableid=0, s=192.168.121.40 (local), d=192.168.121.40
```

```
(FortyGigabitEthernet1/0/1), routed via RIB
*Sep 6 14:45:45.178 IST: IP: s=192.168.121.40 (local), d=192.168.121.40 (FortyGigabitEthernet1/0/1),
len 40, sending
*Sep 6 14:45:45.178 IST: IP: s=192.168.121.40 (local), d=192.168.121.40 (FortyGigabitEthernet1/0/1),
len 40, output feature, NAT Inside(8), rtype 1, forus FALSE, sendself FALSE, mtu 0, fwdchk FALSE
*Sep 6 14:45:45.178 IST: IP: tableid=0, s=192.168.121.40 (FortyGigabitEthernet1/0/1), d=192.168.121.40
(FortyGigabitEthernet1/0/1), routed via RIB
*Sep 6 14:45:45.178 IST: IP: tableid=0, s=192.168.121.40 (FortyGigabitEthernet1/0/1), d=192.168.121.40
(FortyGigabitEthernet1/0/1), routed via RIB
*Sep 6 14:45:45.178 IST: IP: tableid=0, s=192.168.121.40 (FortyGigabitEthernet1/0/1), d=192.168.121.40
(FortyGigabitEthernet1/0/1), routed via RIB
*Sep 6 14:45:45.178 IST: IP: s=192.168.121.40 (local), d=192.168.121.40, len 40, local feature,
feature skipped, NAT(2), rtype 0, forus FALSE, sendself FALSE, mtu 0, fwdchk FALSE
*Sep 6 14:45:45.178 IST: IP: tableid=0, s=192.168.121.40 (local), d=192.168.121.40
(FortyGigabitEthernet1/0/1), routed via RIB
*Sep 6 14:45:45.178 IST: IP: s=192.168.121.40 (local), d=192.168.121.40 (FortyGigabitEthernet1/0/1),
len 40, sending
*Sep 6 14:45:45.178 IST: IP: s=192.168.121.40 (local), d=192.168.121.40 (FortyGigabitEthernet1/0/1),
len 40, output feature, NAT Inside(8), rtype 1, forus FALSE, sendself FALSE, mtu 0, fwdchk FALSE
*Sep 6 14:45:45.179 IST: IP: tableid=0, s=192.168.121.40 (FortyGigabitEthernet1/0/1), d=192.168.121.40
(FortyGigabitEthernet1/0/1), routed via RIB
*Sep 6 14:45:45.179 IST: IP: s=192.168.121.40 (local), d=192.168.121.40, len 40, local feature,
feature skipped, NAT(2), rtype 0, forus FALSE, sendself FALSE, mtu 0, fwdchk FALSE
*Sep 6 14:45:45.179 IST: IP: tableid=0, s=192.168.121.40 (local), d=192.168.121.40
(FortyGigabitEthernet1/0/1), routed via RIB
*Sep 6 14:45:45.179 IST: IP: s=192.168.121.40 (local), d=192.168.121.40 (FortyGigabitEthernet1/0/1),
len 40, sending
*Sep 6 14:45:45.179 IST: IP: s=192.168.121.40 (local), d=192.168.121.40 (FortyGigabitEthernet1/0/1),
len 40, output feature, NAT Inside(8), rtype 1, forus FALSE, sendself FALSE, mtu 0, fwdchk FALSE
*Sep 6 14:45:45.179 IST: IP: tableid=0, s=192.168.121.40 (FortyGigabitEthernet1/0/1), d=192.168.121.40
(FortyGigabitEthernet1/0/1), routed via RIB
```

Configuration Examples for X.509v3 Certificates for SSH Authentication

Example: Configuring Digital Certificates for Server Authentication

```
Switch> enable
Switch# configure terminal
Switch(config)# ip ssh server algorithm hostkey x509v3-ssh-rsa
Switch(config)# ip ssh server certificate profile
Switch(ssh-server-cert-profile)# server
Switch(ssh-server-cert-profile-server)# trustpoint sign trust1
Switch(ssh-server-cert-profile-server)# exit
```

Example: Configuring Digital Certificate for User Authentication

```
Switch> enable
Switch# configure terminal
```

```
Switch(config)# ip ssh server algorithm authentication publickey
Switch(config)# ip ssh server algorithm publickey x509v3-ssh-rsa
Switch(config)# ip ssh server certificate profile
Switch(ssh-server-cert-profile)# user
Switch(ssh-server-cert-profile-user)# trustpoint verify trust2
Switch(ssh-server-cert-profile-user)# end
```

Additional References for X.509v3 Certificates for SSH Authentication

Related Documents

Related Topic	Document Title
PKI configuration	Configuring and Managing a Cisco IOS Certificate Server for PKI Deployment

Technical Assistance

Description	Link
<p>The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies.</p> <p>To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds.</p> <p>Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.</p>	http://www.cisco.com/support

Feature Information for X.509v3 Certificates for SSH Authentication

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Table 130: Feature Information for X509v3 Certificates for SSH Authentication

Feature Name	Releases	Feature Information
X.509v3 Certificates for SSH Authentication	Cisco IOS 15.2(4)E1	<p>The X.509v3 Certificates for SSH Authentication feature uses the X5.09v3 digital certificates in server and user authentication at the SSH server side.</p> <p>The following commands were introduced or modified: ip ssh server algorithm hostkey, ip ssh server algorithm authentication, and ip ssh server certificate profile.</p> <p>This feature was implemented on the following platforms:</p> <ul style="list-style-type: none"> • Catalyst 2960C, 2960CX, 2960P, 2960X, and 2960XR Series Switches • Catalyst 3560CX and 3560X Series Switches • Catalyst 3750X Series Switches • Catalyst 4500E Sup7-E, Sup7L-E, Sup8-E, and 4500X Series Switches • Catalyst 4900M, 4900F-E Series Switches



CHAPTER 65

Configuring Secure Socket Layer HTTP

This feature provides Secure Socket Layer (SSL) version 3.0 support for the HTTP 1.1 server and HTTP 1.1 client within Cisco IOS software. SSL provides server authentication, encryption, and message integrity to allow secure HTTP communications. SSL also provides HTTP client authentication. HTTP over SSL is abbreviated as HTTPS.

- [Information About Secure Socket Layer HTTP](#), on page 1283
- [Monitoring Secure HTTP Server and Client Status](#), on page 1293
- [Configuration Examples for Secure Socket Layer HTTP](#), on page 1294
- [Additional References for Secure Socket Layer HTTP](#), on page 1295
- [Feature Information for Secure Socket Layer HTTP](#), on page 1295
- [Glossary](#), on page 1295

Information About Secure Socket Layer HTTP

Secure HTTP Servers and Clients Overview

On a secure HTTP connection, data to and from an HTTP server is encrypted before being sent over the Internet. HTTP with SSL encryption provides a secure connection to allow such functions as configuring a switch from a Web browser. Cisco's implementation of the secure HTTP server and secure HTTP client uses an implementation of SSL Version 3.0 with application-layer encryption. HTTP over SSL is abbreviated as HTTPS; the URL of a secure connection begins with `https://` instead of `http://`.



Note SSL evolved into Transport Layer Security (TLS) in 1999, but is still used in this particular context.

The primary role of the HTTP secure server (the switch) is to listen for HTTPS requests on a designated port (the default HTTPS port is 443) and pass the request to the HTTP 1.1 Web server. The HTTP 1.1 server processes requests and passes responses (pages) back to the HTTP secure server, which, in turn, responds to the original request.

The primary role of the HTTP secure client (the web browser) is to respond to Cisco IOS application requests for HTTPS User Agent services, perform HTTPS User Agent services for the application, and pass the response back to the application.

Certificate Authority Trustpoints

Certificate authorities (CAs) manage certificate requests and issue certificates to participating network devices. These services provide centralized security key and certificate management for the participating devices. Specific CA servers are referred to as *trustpoints*.

When a connection attempt is made, the HTTPS server provides a secure connection by issuing a certified X.509v3 certificate, obtained from a specified CA trustpoint, to the client. The client (usually a Web browser), in turn, has a public key that allows it to authenticate the certificate.

For secure HTTP connections, we highly recommend that you configure a CA trustpoint. If a CA trustpoint is not configured for the device running the HTTPS server, the server certifies itself and generates the needed RSA key pair. Because a self-certified (self-signed) certificate does not provide adequate security, the connecting client generates a notification that the certificate is self-certified, and the user has the opportunity to accept or reject the connection. This option is useful for internal network topologies (such as testing).

If you do not configure a CA trustpoint, when you enable a secure HTTP connection, either a temporary or a persistent self-signed certificate for the secure HTTP server (or client) is automatically generated.

- If the switch is not configured with a hostname and a domain name, a temporary self-signed certificate is generated. If the switch reboots, any temporary self-signed certificate is lost, and a new temporary new self-signed certificate is assigned.
- If the switch has been configured with a host and domain name, a persistent self-signed certificate is generated. This certificate remains active if you reboot the switch or if you disable the secure HTTP server so that it will be there the next time you re-enable a secure HTTP connection.



Note The certificate authorities and trustpoints must be configured on each device individually. Copying them from other devices makes them invalid on the switch.

When a new certificate is enrolled, the new configuration change is not applied to the HTTPS server until the server is restarted. You can restart the server using either the CLI or by physical reboot. On restarting the server, the switch starts using the new certificate.

If a self-signed certificate has been generated, this information is included in the output of the **show running-config** privileged EXEC command. This is a partial sample output from that command displaying a self-signed certificate.

```
Device# show running-config
Building configuration...

<output truncated>

crypto pki trustpoint TP-self-signed-3080755072
  enrollment selfsigned
  subject-name cn=IOS-Self-Signed-Certificate-3080755072
  revocation-check none
  rsakeypair TP-self-signed-3080755072
  !
!
crypto ca certificate chain TP-self-signed-3080755072
certificate self-signed 01
  3082029F 30820208 A0030201 02020101 300D0609 2A864886 F70D0101 04050030
  59312F30 2D060355 04031326 494F532D 53656C66 2D536967 6E65642D 43657274
  69666963 6174652D 33303830 37353530 37323126 30240609 2A864886 F70D0109
```

```
02161743 45322D33 3535302D 31332E73 756D6D30 342D3335 3530301E 170D3933
30333031 30303030 35395A17 0D323030 31303130 30303030 305A3059 312F302D
```

<output truncated>

You can remove this self-signed certificate by disabling the secure HTTP server and entering the **no crypto pki trustpoint TP-self-signed-30890755072** global configuration command. If you later re-enable a secure HTTP server, a new self-signed certificate is generated.



Note The values that follow *TP self-signed* depend on the serial number of the device.

You can use an optional command (**ip http secure-client-auth**) to allow the HTTPS server to request an X.509v3 certificate from the client. Authenticating the client provides more security than server authentication by itself.

CipherSuites

A CipherSuite specifies the encryption algorithm and the digest algorithm to use on a SSL connection. When connecting to the HTTPS server, the client Web browser offers a list of supported CipherSuites, and the client and server negotiate the best encryption algorithm to use from those on the list that are supported by both. For example, Netscape Communicator 4.76 supports U.S. security with RSA Public Key Cryptography, MD2, MD5, RC2-CBC, RC4, DES-CBC, and DES-EDE3-CBC.

For the best possible encryption, you should use a client browser that supports 128-bit encryption, such as Microsoft Internet Explorer Version 5.5 (or later) or Netscape Communicator Version 4.76 (or later). The SSL_RSA_WITH_DES_CBC_SHA CipherSuite provides less security than the other CipherSuites, as it does not offer 128-bit encryption.

The more secure and more complex CipherSuites require slightly more processing time. This list defines the CipherSuites supported by the switch and ranks them from fastest to slowest in terms of router processing load (speed):

1. SSL_RSA_WITH_DES_CBC_SHA—RSA key exchange (RSA Public Key Cryptography) with DES-CBC for message encryption and SHA for message digest
2. SSL_RSA_WITH_NULL_SHA key exchange with NULL for message encryption and SHA for message digest (only for SSL 3.0).
3. SSL_RSA_WITH_NULL_MD5 key exchange with NULL for message encryption and MD5 for message digest (only for SSL 3.0).
4. SSL_RSA_WITH_RC4_128_MD5—RSA key exchange with RC4 128-bit encryption and MD5 for message digest
5. SSL_RSA_WITH_RC4_128_SHA—RSA key exchange with RC4 128-bit encryption and SHA for message digest
6. SSL_RSA_WITH_3DES_EDE_CBC_SHA—RSA key exchange with 3DES and DES-EDE3-CBC for message encryption and SHA for message digest
7. SSL_RSA_WITH_AES_128_CBC_SHA—RSA key exchange with AES 128-bit encryption and SHA for message digest (only for SSL 3.0).

8. `SSL_RSA_WITH_AES_256_CBC_SHA`—RSA key exchange with AES 256-bit encryption and SHA for message digest (only for SSL 3.0).
9. `SSL_RSA_WITH_DHE_AES_128_CBC_SHA`—RSA key exchange with AES 128-bit encryption and SHA for message digest (only for SSL 3.0).
10. `SSL_RSA_WITH_DHE_AES_256_CBC_SHA`—RSA key exchange with AES 256-bit encryption and SHA for message digest (only for SSL 3.0).



Note The latest versions of Chrome do not support the four original cipher suites, thus disallowing access to both web GUI and guest portals.

RSA (in conjunction with the specified encryption and digest algorithm combinations) is used for both key generation and authentication on SSL connections. This usage is independent of whether or not a CA trustpoint is configured.

Default SSL Configuration

The standard HTTP server is enabled.

SSL is enabled.

No CA trustpoints are configured.

No self-signed certificates are generated.

SSL Configuration Guidelines

When SSL is used in a switch cluster, the SSL session terminates at the cluster commander. Cluster member switches must run standard HTTP.

Before you configure a CA trustpoint, you should ensure that the system clock is set. If the clock is not set, the certificate is rejected due to an incorrect date.

In a switch stack, the SSL session terminates at the active switch.

How to Configure Secure Socket Layer HTTP

Configuring the Secure HTTP Server

Beginning in privileged EXEC mode, follow these steps to configure a secure HTTP server:

Before you begin

If you are using a certificate authority for certification, you should use the previous procedure to configure the CA trustpoint on the switch before enabling the HTTP server. If you have not configured a CA trustpoint, a self-signed certificate is generated the first time that you enable the secure HTTP server. After you have configured the server, you can configure options (path, access list to apply, maximum number of connections, or timeout policy) that apply to both standard and secure HTTP servers.

To verify the secure HTTP connection by using a Web browser, enter `https://URL`, where the URL is the IP address or hostname of the server switch. If you configure a port other than the default port, you must also specify the port number after the URL. For example:



Note AES256_SHA2 is not supported.

```
https://209.165.129:1026
```

or

```
https://host.domain.com:1026
```

The existing **ip http access-class** *access-list-number* command for specifying the access-list (Only IPv4 ACLs) is going to be deprecated. You can still use this command to specify an access list to allow access to the HTTP server. Two new commands have been introduced to enable support for specifying IPv4 and IPv6 ACLs. These are **ip http access-class ipv4** *access-list-name* | *access-list-number* for specifying IPv4 ACLs and **ip http access-class ipv6** *access-list-name* for specifying IPv6 ACLs. We recommend using the new CLI to avoid receiving warning messages.

Note the following considerations for specifying access-lists:

- If you specify an access-list that does not exist, the configuration takes place but you receive the below warning message:

```
ACL being attached does not exist, please configure it
```

- If you use the **ip http access-class** command for specifying an access-list for the HTTP server, the below warning message appears:

```
This CLI will be deprecated soon, Please use new CLI ip http
access-class ipv4/ipv6 <access-list-name>| <access-list-number>
```

- If you use **ip http access-class ipv4** *access-list-name* | *access-list-number* or **ip http access-class ipv6** *access-list-name*, and an access-list was already configured using **ip http access-class**, the below warning message appears:

```
Removing ip http access-class <access-list-number>
```

ip http access-class *access-list-number* and **ip http access-class ipv4** *access-list-name* | *access-list-number* share the same functionality. Each command overrides the configuration of the previous command. The following combinations between the configuration of the two commands explain the effect on the running configuration:

- If **ip http access-class** *access-list-number* is already configured and you try to configure using **ip http access-class ipv4** *access-list-number* command, the configuration of **ip http access-class** *access-list-number* will be removed and the configuration of **ip http access-class ipv4** *access-list-number* will be added to the running configuration.
- If **ip http access-class** *access-list-number* is already configured and you try to configure using **ip http access-class ipv4** *access-list-name* command, the configuration of **ip http access-class** *access-list-number* will be removed and the configuration of **ip http access-class ipv4** *access-list-name* will be added to the running configuration.

- If **ip http access-class ipv4** *access-list-number* is already configured and you try to configure using **ip http access-class** *access-list-name*, the configuration of **ip http access-class ipv4** *access-list-number* will be removed from configuration and the configuration of **ip http access-class** *access-list-name* will be added to the running configuration.
- If **ip http access-class ipv4** *access-list-name* is already configured and you try to configure using **ip http access-class** *access-list-number*, the configuration of **ip http access-class ipv4** *access-list-name* will be removed from the configuration and the configuration of **ip http access-class** *access-list-number* will be added to the running configuration.

SUMMARY STEPS

1. **show ip http server status**
2. **configure terminal**
3. **ip http secure-server**
4. **ip http secure-port** *port-number*
5. **ip http secure-ciphersuite** {[3des-edc-cbc-sha] [rc4-128-md5] [rc4-128-sha] [des-cbc-sha]}
6. **ip http secure-client-auth**
7. **ip http secure-trustpoint** *name*
8. **ip http path** *path-name*
9. **ip http access-class** *access-list-number*
10. **ip http access-class** { **ipv4** {*access-list-number* | *access-list-name*} | **ipv6** {*access-list-name*} }
11. **ip http max-connections** *value*
12. **ip http timeout-policy** **idle** *seconds* **life** *seconds* **requests** *value*
13. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	show ip http server status Example: Device# show ip http server status	(Optional) Displays the status of the HTTP server to determine if the secure HTTP server feature is supported in the software. You should see one of these lines in the output: HTTP secure server capability: Present or HTTP secure server capability: Not present
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.

	Command or Action	Purpose
Step 3	ip http secure-server Example: <pre>Device(config)# ip http secure-server</pre>	Enables the HTTPS server if it has been disabled. The HTTPS server is enabled by default.
Step 4	ip http secure-port <i>port-number</i> Example: <pre>Device(config)# ip http secure-port 443</pre>	(Optional) Specifies the port number to be used for the HTTPS server. The default port number is 443. Valid options are 443 or any number in the range 1025 to 65535.
Step 5	ip http secure-ciphersuite {[3des-ede-cbc-sha] [rc4-128-md5] [rc4-128-sha] [des-cbc-sha]} Example: <pre>Device(config)# ip http secure-ciphersuite rc4-128-md5</pre>	(Optional) Specifies the CipherSuites (encryption algorithms) to be used for encryption over the HTTPS connection. If you do not have a reason to specify a particular CipherSuite, you should allow the server and client to negotiate a CipherSuite that they both support. This is the default.
Step 6	ip http secure-client-auth Example: <pre>Device(config)# ip http secure-client-auth</pre>	(Optional) Configures the HTTP server to request an X.509v3 certificate from the client for authentication during the connection process. The default is for the client to request a certificate from the server, but the server does not attempt to authenticate the client.
Step 7	ip http secure-trustpoint <i>name</i> Example: <pre>Device(config)# ip http secure-trustpoint your_trustpoint</pre>	Specifies the CA trustpoint to use to get an X.509v3 security certificate and to authenticate the client certificate connection. Note Use of this command assumes you have already configured a CA trustpoint according to the previous procedure.
Step 8	ip http path <i>path-name</i> Example: <pre>Device(config)# ip http path /your_server:80</pre>	(Optional) Sets a base HTTP path for HTML files. The path specifies the location of the HTTP server files on the local system (usually located in system flash memory).
Step 9	ip http access-class <i>access-list-number</i> Example: <pre>Device(config)# ip http access-class 2</pre>	(Optional) Specifies an access list to use to allow access to the HTTP server.
Step 10	ip http access-class { ipv4 { <i>access-list-number</i> <i>access-list-name</i> } ipv6 { <i>access-list-name</i> } } Example:	(Optional) Specifies an access list to use to allow access to the HTTP server.

	Command or Action	Purpose
	Device(config)# ip http access-class ipv4 4	
Step 11	ip http max-connections <i>value</i> Example: Device(config)# ip http max-connections 4	(Optional) Sets the maximum number of concurrent connections that are allowed to the HTTP server. We recommend that the value be at least 10 and not less. This is required for the UI to function as expected.
Step 12	ip http timeout-policy idle <i>seconds</i> life <i>seconds</i> requests <i>value</i> Example: Device(config)# ip http timeout-policy idle 120 life 240 requests 1	(Optional) Specifies how long a connection to the HTTP server can remain open under the defined circumstances: <ul style="list-style-type: none"> • idle—the maximum time period when no data is received or response data cannot be sent. The range is 1 to 600 seconds. The default is 180 seconds (3 minutes). • life—the maximum time period from the time that the connection is established. The range is 1 to 86400 seconds (24 hours). The default is 180 seconds. • requests—the maximum number of requests processed on a persistent connection. The maximum value is 86400. The default is 1.
Step 13	end Example: Device(config)# end	Returns to privileged EXEC mode.

Configuring the Secure HTTP Client

Beginning in privileged EXEC mode, follow these steps to configure a secure HTTP client:

Before you begin

The standard HTTP client and secure HTTP client are always enabled. A certificate authority is required for secure HTTP client certification. This procedure assumes that you have previously configured a CA trustpoint on the switch. If a CA trustpoint is not configured and the remote HTTPS server requires client authentication, connections to the secure HTTP client fail.

SUMMARY STEPS

1. **configure terminal**
2. **ip http client secure-trustpoint** *name*
3. **ip http client secure-ciphersuite** {[3des-ede-cbc-sha] [rc4-128-md5] [rc4-128-sha] [des-cbc-sha]}
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode.
Step 2	ip http client secure-trustpoint <i>name</i> Example: Device(config)# <code>ip http client secure-trustpoint your_trustpoint</code>	(Optional) Specifies the CA trustpoint to be used if the remote HTTP server requests client authentication. Using this command assumes that you have already configured a CA trustpoint by using the previous procedure. The command is optional if client authentication is not needed or if a primary trustpoint has been configured.
Step 3	ip http client secure-ciphersuite {[3des-ede-cbc-sha] [rc4-128-md5] [rc4-128-sha] [des-cbc-sha]} Example: Device(config)# <code>ip http client secure-ciphersuite rc4-128-md5</code>	(Optional) Specifies the CipherSuites (encryption algorithms) to be used for encryption over the HTTPS connection. If you do not have a reason to specify a particular CipherSuite, you should allow the server and client to negotiate a CipherSuite that they both support. This is the default.
Step 4	end Example: Device(config)# <code>end</code>	Returns to privileged EXEC mode.

Configuring a CA Trustpoint

For secure HTTP connections, we recommend that you configure an official CA trustpoint. A CA trustpoint is more secure than a self-signed certificate.

Beginning in privileged EXEC mode, follow these steps to configure a CA Trustpoint:

SUMMARY STEPS

1. **configure terminal**
2. **hostname *hostname***
3. **ip domain-name *domain-name***
4. **crypto key generate rsa**
5. **crypto ca trustpoint *name***
6. **enrollment url *url***
7. **enrollment http-proxy *host-name port-number***
8. **crl query *url***
9. **primary *name***
10. **exit**

11. `crypto ca authentication name`
12. `crypto ca enroll name`
13. `end`

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode.
Step 2	hostname <i>hostname</i> Example: Device(config)# <code>hostname your_hostname</code>	Specifies the hostname of the switch (required only if you have not previously configured a hostname). The hostname is required for security keys and certificates.
Step 3	ip domain-name <i>domain-name</i> Example: Device(config)# <code>ip domain-name your_domain</code>	Specifies the IP domain name of the switch (required only if you have not previously configured an IP domain name). The domain name is required for security keys and certificates.
Step 4	crypto key generate rsa Example: Device(config)# <code>crypto key generate rsa</code>	(Optional) Generates an RSA key pair. RSA key pairs are required before you can obtain a certificate for the switch. RSA key pairs are generated automatically. You can use this command to regenerate the keys, if needed.
Step 5	crypto ca trustpoint <i>name</i> Example: Device(config)# <code>crypto ca trustpoint your_trustpoint</code>	Specifies a local configuration name for the CA trustpoint and enter CA trustpoint configuration mode.
Step 6	enrollment url <i>url</i> Example: Device(ca-trustpoint)# <code>enrollment url http://your_server:80</code>	Specifies the URL to which the switch should send certificate requests.
Step 7	enrollment http-proxy <i>host-name port-number</i> Example: Device(ca-trustpoint)# <code>enrollment http-proxy</code>	(Optional) Configures the switch to obtain certificates from the CA through an HTTP proxy server. <ul style="list-style-type: none"> • For <i>host-name</i> , specify the proxy server used to get the CA.

	Command or Action	Purpose
	<code>your_host 49</code>	<ul style="list-style-type: none"> For <i>port-number</i>, specify the port number used to access the CA.
Step 8	crl query url Example: <pre>Device(ca-trustpoint)# crl query ldap://your_host:49</pre>	Configures the switch to request a certificate revocation list (CRL) to ensure that the certificate of the peer has not been revoked.
Step 9	primary name Example: <pre>Device(ca-trustpoint)# primary your_trustpoint</pre>	(Optional) Specifies that the trustpoint should be used as the primary (default) trustpoint for CA requests. <ul style="list-style-type: none"> For <i>name</i>, specify the trustpoint that you just configured.
Step 10	exit Example: <pre>Device(ca-trustpoint)# exit</pre>	Exits CA trustpoint configuration mode and return to global configuration mode.
Step 11	crypto ca authentication name Example: <pre>Device(config)# crypto ca authentication your_trustpoint</pre>	Authenticates the CA by getting the public key of the CA. Use the same name used in Step 5.
Step 12	crypto ca enroll name Example: <pre>Device(config)# crypto ca enroll your_trustpoint</pre>	Obtains the certificate from the specified CA trustpoint. This command requests a signed certificate for each RSA key pair.
Step 13	end Example: <pre>Device(config)# end</pre>	Returns to privileged EXEC mode.

Monitoring Secure HTTP Server and Client Status

To monitor the SSL secure server and client status, use the privileged EXEC commands in the following table.

Table 131: Commands for Displaying the SSL Secure Server and Client Status

Command	Purpose
<code>show ip http client secure status</code>	Shows the HTTP secure client configuration.

Command	Purpose
show ip http server secure status	Shows the HTTP secure server configuration.
show running-config	Shows the generated self-signed certificate for secure HTTP connections.

Configuration Examples for Secure Socket Layer HTTP

Example: Configuring Secure Socket Layer HTTP

The following example shows a configuration session in which the secure HTTP server is enabled, the port for the secure HTTP server is configured as 1025, and the remote CA trustpoint server “CA-trust-local” is used for certification.

```
Device# show ip http server status

HTTP server status: Disabled
HTTP server port: 80
HTTP server authentication method: enable
HTTP server access class: 0
HTTP server base path:
Maximum number of concurrent server connections allowed: 5
Server idle time-out: 600 seconds
Server life time-out: 600 seconds
Maximum number of requests allowed on a connection: 1
HTTP secure server capability: Present
HTTP secure server status: Disabled
HTTP secure server port: 443
HTTP secure server ciphersuite: 3des-ede-cbc-sha des-cbc-sha rc4-128-md5 rc4-12a
HTTP secure server client authentication: Disabled
HTTP secure server trustpoint:
```

```
Device# configure terminal
Device(config)# ip http secure-server
Device(config)# ip http client secure-trustpoint CA-trust-local
Device(config)# ip http secure-port 1024
Invalid secure port value.
Device(config)# ip http secure-port 1025
Device(config)# ip http secure-ciphersuite rc4-128-sha rc4-128-md5
Device(config)# end
```

```
Device# show ip http serversecure status

HTTP secure server status: Enabled
HTTP secure server port: 1025
HTTP secure server ciphersuite: rc4-128-md5 rc4-128-sha
HTTP secure server client authentication: Disabled
HTTP secure server trustpoint: CA-trust-local
```

In the following example, the CA trustpoint CA-trust-local is specified, and the HTTPS client is configured to use this trustpoint for client authentication requests:


```

Device# config terminal
Device(config)# crypto ca trustpoint CA-trust-local
Device(ca-trustpoint)# enrollment url http://example.com
Device(ca-trustpoint)# crl query ldap://example.com
Device(ca-trustpoint)# primary
Device(ca-trustpoint)# exit
Device(config)# ip http client secure-trustpoint CA-trust-local
Device(config)# end
Device# copy running-config startup-config

```

Additional References for Secure Socket Layer HTTP

Related Documents

Related Topic	Document Title
Cisco IOS commands	Cisco IOS Master Command List, All Releases

Technical Assistance

Description	Link
<p>The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies.</p> <p>To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds.</p> <p>Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.</p>	http://www.cisco.com/support

Feature Information for Secure Socket Layer HTTP

Release	Feature Information
Cisco IOS Release 15.0(2)EX1	This feature was introduced.

Glossary

RSA—RSA is a widely used Internet encryption and authentication system that uses public and private keys for encryption and decryption. The RSA algorithm was invented in 1978 by Ron Rivest, Adi Shamir, and Leonard Adleman. The abbreviation RSA comes from the first letter of the last names of the three original

developers. The RSA algorithm is included in many applications, such as the web browsers from Microsoft and Netscape. The RSA encryption system is owned by RSA Security.

SHA—The Secure Hash Algorithm. SHA was developed by NIST and is specified in the Secure Hash Standard (SHS, FIPS 180). Often used as an alternative to Digest 5 algorithm.

signatures, digital—In the context of SSL, “signing” means to encrypt with a private key. In digital signing, one-way hash functions are used as input for a signing algorithm. In RSA signing, a 36-byte structure of two hashes (one SHA and one MD5) is signed (encrypted with the private key).

SSL 3.0—Secure Socket Layer version 3.0. SSL is a security protocol that provides communications privacy over the Internet. The protocol allows client and server applications to communicate in a way that is designed to prevent eavesdropping, tampering, or message forgery. SSL uses a program layer located between the Internet’s HTTP and TCP layers. SSL is included as part of most web server products and as part of most Internet browsers. The SSL 3.0 specification can be found at <https://tools.ietf.org/html/rfc6101>.



CHAPTER 66

Access Control List Overview

Access lists filter network traffic by controlling the forwarding or blocking of packets at the interface of a device. A device examines each packet to determine whether to forward or drop that packet, based on the criteria specified in access lists.

The criteria that can be specified in an access list include the source address of the traffic, the destination address of the traffic, and the upper-layer protocol.



Note Some users might successfully evade basic access lists because these lists require no authentication.

- [Information About Access Control Lists, on page 1297](#)

Information About Access Control Lists

Definition of an Access List

An access list is a sequential list consisting of at least one **permit** statement and possibly one or more **deny** statements. In the case of IP access lists, the statements can apply to IP addresses, upper-layer IP protocols, or other fields in IP packets. The access list is identified and referenced by a name or a number. Access list acts as a packet filter, filtering packets based on the criteria defined in the access list.

An access list may be configured, but it does not take effect until the access list is either applied to an interface, a virtual terminal line (vty), or referenced by some command that accepts an access list. Multiple commands can reference the same access list.

The following configuration example shows how to create an IP access list named `branchoffices`. The ACL is applied to serial interface 0 on incoming packets. No sources other than those on the networks specified by each source address and mask pair can access this interface. The destinations for packets coming from sources on network 172.20.7.0 are unrestricted. The destination for packets coming from sources on network 172.29.2.0 must be 172.25.5.4.

```
ip access-list extended branchoffices
 10 permit 172.20.7.0 0.0.0.3 any
 20 permit 172.29.2.0 0.0.0.255 host 172.25.5.4
!
interface serial 0
 ip access-group branchoffices in
```

Functions of an Access Control List

There are many reasons to configure access lists; for example, to restrict contents of routing updates or to provide traffic flow control. One of the most important reasons to configure access lists is to provide security for your network, which is the focus of this module.

Use access lists to provide a basic level of security for accessing your network. If you do not configure access lists on your device, all packets passing through the device are allowed access to all parts of your network.

Access lists can allow a host to access a part of your network and prevent another host from accessing the same area. In the figure below, Host A is allowed to access the Human Resources network, but Host B is prevented from accessing the Human Resources network.

You can also use access lists to define the type of traffic that is forwarded or blocked at device interfaces. For example, you can permit e-mail traffic to be routed but at the same time block all Telnet traffic.

Purpose of IP Access Lists

Access lists perform packet filtering to control which packets move through the network and where. Such control can help limit network traffic and restrict the access of users and devices to the network. Access lists have many uses, and therefore many commands accept a reference to an access list in their command syntax. Access lists can be used to do the following:

- Filter incoming packets on an interface.
- Filter outgoing packets on an interface.
- Restrict the contents of routing updates.
- Limit debug output based on an address or protocol.
- Control virtual terminal line access.
- Identify or classify traffic for advanced features, such as congestion avoidance, congestion management, and priority and custom queuing.
- Trigger dial-on-demand routing (DDR) calls.

Reasons to Configure ACLs

There are many reasons to configure access lists; for example, you can use access lists to restrict contents of switching updates or to provide traffic flow control. One of the most important reasons to configure access lists is to provide a basic level of security for your network by controlling access to it. If you do not configure access lists on your device, all packets passing through the device could be allowed onto all parts of your network.

An access list can allow one host to access a part of your network and prevent another host from accessing the same area. For example, by applying an appropriate access list to interfaces of a device, Host A is allowed to access the human resources network and Host B is prevented from accessing the human resources network.

You can use access lists on a device that is positioned between two parts of your network, to control traffic entering or exiting a specific part of your internal network.

To provide some security benefits of access lists, you should at least configure access lists on border devices—devices located at the edges of your networks. Such an access list provides a basic buffer from the

outside network or from a less controlled area of your own network into a more sensitive area of your network. On these border devices, you should configure access lists for each network protocol configured on the device interfaces. You can configure access lists so that inbound traffic or outbound traffic or both are filtered on an interface.

Access lists are defined on a per-protocol basis. In other words, you should define access lists for every protocol enabled on an interface if you want to control traffic flow for that protocol.

Software Processing of an Access List

The following general steps describe how the an access list is processed when it is applied to an interface, a vty, or referenced by any command. These steps apply to an access list that has 13 or fewer access list entries.

- The software receives an IP packet and tests parts of each packet being filtered against the conditions in the access list, one condition (**permit** or **deny** statement) at a time. For example, the software tests the source and destination addresses of the packet against the source and destination addresses in a **permit** or **deny** statement.
- If a packet does not match an access list statement, the packet is then tested against the next statement in the list.
- If a packet and an access list statement match, the rest of the statements in the list are skipped and the packet is permitted or denied as specified in the matched statement. The first entry that the packet matches determines whether the software permits or denies the packet. That is, after the first match, no subsequent entries are considered.
- If the access list denies a packet, the software discards the packet and returns an Internet Control Message Protocol (ICMP) Host Unreachable message.
- If no conditions match, the software drops the packet. This is because each access list ends with an unwritten, implicit **deny** statement. That is, if the packet has not been permitted by the time it was tested against each statement, it is denied.

An access list with more than 13 entries is processed using a trie-based lookup algorithm. This process will happen automatically; it does not need to be configured.

Access List Rules

The following rules apply to access control lists (ACLs):

- Only one access list per interface, per protocol, and per direction is allowed.
- An access list must contain at least one **permit** statement or all packets are denied entry into the network.
- The order in which access list conditions or match criteria are configured is important. While deciding whether to forward or block a packet, Cisco software tests the packet against each criteria statement in the order in which these statements are created. After a match is found, no more criteria statements are checked. The same **permit** or **deny** statements specified in a different order can result in a packet being passed under one circumstance and denied in another circumstance.
- If an access list is referenced by a name, but the access list does not exist, all packets pass. An interface or command with an empty access list applied to it permits all traffic into the network.
- Standard access lists and extended access lists cannot have the same name.

- Inbound access lists process packets before packets are sent to an outbound interface. Inbound access lists that have filtering criteria that deny packet access to a network saves the overhead of a route lookup. Packets that are permitted access to a network based on the configured filtering criteria are processed for routing. For inbound access lists, when you configure a **permit** statement, packets are processed after they are received, and when you configure a **deny** statement, packets are discarded.
- Outbound access lists process packets before they leave the device. Incoming packets are routed to the outbound interface and then processed by the outbound access list. For outbound access lists, when you configure a **permit** statement, packets are sent to the output buffer, and when you configure a **deny** statement, packets are discarded.
- An access list can control traffic arriving at a device or leaving a device, but not traffic originating at a device.

Helpful Hints for Creating IP Access Lists

The following tips will help you avoid unintended consequences and help you create more efficient access lists.

- Create the access list before applying it to an interface (or elsewhere), because if you apply a nonexistent access list to an interface and then proceed to configure the access list, the first statement is put into effect, and the implicit **deny** statement that follows could cause you immediate access problems.
- Another reason to configure an access list before applying it is because an interface with an empty access list applied to it permits all traffic.
- All access lists need at least one **permit** statement; otherwise, all packets are denied and no traffic passes.
- Because the software stops testing conditions after it encounters the first match (to either a **permit** or **deny** statement), you will reduce processing time and resources if you put the statements that packets are most likely to match at the beginning of the access list. Place more frequently occurring conditions before less frequent conditions.
- Organize your access list so that more specific references in a network or subnet appear before more general ones.
- Use the statement **permit any any** if you want to allow all other packets not already denied. Using the statement **permit any any** in effect avoids denying all other packets with the implicit deny statement at the end of an access list. Do not make your first access list entry **permit any any** because all traffic will get through; no packets will reach the subsequent testing. In fact, once you specify **permit any any**, all traffic not already denied will get through.
- Although all access lists end with an implicit **deny** statement, we recommend use of an explicit **deny** statement (for example, **deny ip any any**). On most platforms, you can display the count of packets denied by issuing the **show access-list** command, thus finding out more information about who your access list is disallowing. Only packets denied by explicit **deny** statements are counted, which is why the explicit **deny** statement will yield more complete data for you.
- While you are creating an access list or after it is created, you might want to delete an entry.
 - You cannot delete an entry from a numbered access list; trying to do so will delete the entire access list. If you need to delete an entry, you need to delete the entire access list and start over.
 - You can delete an entry from a named access list. Use the **no permit** or **no deny** command to delete the appropriate entry.

- In order to make the purpose of individual statements more scannable and easily understood at a glance, you can write a helpful remark before or after any statement by using the **remark** command.
- If you want to deny access to a particular host or network and find out if someone from that network or host is attempting to gain access, include the **log** keyword with the corresponding **deny** statement so that the packets denied from that source are logged for you.
- This hint applies to the placement of your access list. When trying to save resources, remember that an inbound access list applies the filter conditions before the routing table lookup. An outbound access list applies the filter conditions after the routing table lookup.

IP Packet Fields You Can Filter to Control Access

You can use an extended access list to filter on any of the following fields in an IP packet. Source address and destination address are the two most frequently specified fields on which to base an access list:

- Source address--Specifies a source address to control packets coming from certain networking devices or hosts.
- Destination address--Specifies a destination address to control packets being sent to certain networking devices or hosts.
- Protocol--Specifies an IP protocol indicated by the keyword **eigrp**, **gre**, **icmp**, **igmp**, **ip**, **ipinip**, **nos**, **ospf**, **tcp**, or **udp**, or indicated by an integer in the range from 0 to 255 (representing an Internet protocol). If you specify a transport layer protocol (**icmp**, **igmp**, **tcp**, or **udp**), the command has a specific syntax.
 - Ports and non-contiguous ports--Specifies TCP or UDP ports by a port name or port number. The port numbers can be noncontiguous port numbers. Port numbers can be useful to filter Telnet traffic or HTTP traffic, for example.
 - TCP flags--Specifies that packets match any flag or all flags set in TCP packets. Filtering on specific TCP flags can help prevent false synchronization packets.
- IP options--Specifies IP options; one reason to filter on IP options is to prevent routers from being saturated with spurious packets containing them.

Source and Destination Addresses

Source and destination address fields in an IP packet are two typical fields on which to base an access list. Specify source addresses to control the packets being sent from certain networking devices or hosts. Specify destination addresses to control the packets being sent to certain networking devices or hosts.

Wildcard Mask for Addresses in an Access List

Address filtering uses wildcard masking to indicate to the software whether to check or ignore corresponding IP address bits when comparing the address bits in an access list entry to a packet being submitted to the access list. By carefully setting wildcard masks, you can specify one or more IP addresses for permit or deny tests.

Wildcard masking for IP address bits uses the number 1 and the number 0 to specify how the software treats the corresponding IP address bits. A wildcard mask is sometimes referred to as an inverted mask because a 1 and 0 mean the opposite of what they mean in a subnet (network) mask.

- A wildcard mask bit 0 means check the corresponding bit value; they must match.
- A wildcard mask bit 1 means ignore that corresponding bit value; they need not match.

If you do not supply a wildcard mask with a source or destination address in an access list statement, the software assumes an implicit wildcard mask of 0.0.0.0, meaning all values must match.

Unlike subnet masks, which require contiguous bits indicating network and subnet to be ones, wildcard masks allow noncontiguous bits in the mask.

The table below shows examples of IP addresses and masks from an access list, along with the corresponding addresses that are considered a match.

Table 132: Sample IP Addresses, Wildcard Masks, and Match Results

Address	Wildcard Mask	Match Results
0.0.0.0	255.255.255.255	All addresses will match the access list conditions.
172.18.0.0/16	0.0.255.255	Network 172.18.0.0
172.18.5.2/16	0.0.0.0	Only host 172.18.5.2 matches
172.18.8.0	0.0.0.7	Only subnet 172.18.8.0/29 matches
172.18.8.8	0.0.0.7	Only subnet 172.18.8.8/29 matches
172.18.8.15	0.0.0.3	Only subnet 172.18.8.15/30 matches
10.1.2.0	0.0.254.255 (noncontiguous bits in mask)	Matches any even-numbered network in the range of 10.1.2.0 to 10.1.254.0

Access List Sequence Numbers

The ability to apply sequence numbers to IP access list entries simplifies access list changes. Prior to the IP Access List Entry Sequence Numbering feature, there was no way to specify the position of an entry within an access list. If you wanted to insert an entry in the middle of an existing list, all of the entries after the desired position had to be removed, then the new entry was added, and then all the removed entries had to be reentered. This method was cumbersome and error prone.

This feature allows users to add sequence numbers to access list entries and resequence them. When you add a new entry, you specify the sequence number so that it is in a desired position in the access list. If necessary, entries currently in the access list can be resequenced to create room to insert the new entry.

ACL Supported Types

The switch supports IP ACLs and Ethernet (MAC) ACLs:

- IP ACLs filter IPv4 traffic, including TCP, User Datagram Protocol (UDP), Internet Group Management Protocol (IGMP), and Internet Control Message Protocol (ICMP).
- Ethernet ACLs filter non-IP traffic.

This switch also supports quality of service (QoS) classification ACLs.

Supported ACLs

The switch supports three types of ACLs to filter traffic:

- Port ACLs access-control traffic entering a Layer 2 interface. You can apply port ACLs to a Layer 2 interface in each direction to each access list type — IPv4 and MAC.
- Router ACLs access-control routed traffic between VLANs and are applied to Layer 3 interfaces in a specific direction (inbound or outbound).
- VLAN ACLs or VLAN maps access-control all packets (bridged and routed). You can use VLAN maps to filter traffic between devices in the same VLAN. VLAN maps are configured to provide access control based on Layer 3 addresses for IPv4. Unsupported protocols are access-controlled through MAC addresses using Ethernet ACEs. After a VLAN map is applied to a VLAN, all packets (routed or bridged) entering the VLAN are checked against the VLAN map. Packets can either enter the VLAN through a switch port or through a routed port after being routed.

ACL Precedence

When VLAN maps, Port ACLs, and router ACLs are configured on the same switch, the filtering precedence, from greatest to least for ingress traffic is port ACL, VLAN map, and then router ACL. For egress traffic, the filtering precedence is router ACL, VLAN map, and then port ACL.

The following examples describe simple use cases:

- When both an input port ACL and a VLAN map are applied, incoming packets received on ports with a port ACL applied are filtered by the port ACL. Other packets are filtered by the VLAN map
- When an input router ACL and input port ACL exist in a switch virtual interface (SVI), incoming packets received on ports to which a port ACL is applied are filtered by the port ACL. Incoming routed IP packets received on other ports are filtered by the router ACL. Other packets are not filtered.
- When an output router ACL and input port ACL exist in an SVI, incoming packets received on the ports to which a port ACL is applied are filtered by the port ACL. Outgoing routed IP packets are filtered by the router ACL. Other packets are not filtered.
- When a VLAN map, input router ACL, and input port ACL exist in an SVI, incoming packets received on the ports to which a port ACL is applied are only filtered by the port ACL. Incoming routed IP packets received on other ports are filtered by both the VLAN map and the router ACL. Other packets are filtered only by the VLAN map.
- When a VLAN map, output router ACL, and input port ACL exist in an SVI, incoming packets received on the ports to which a port ACL is applied are only filtered by the port ACL. Outgoing routed IP packets are filtered by both the VLAN map and the router ACL. Other packets are filtered only by the VLAN map.

Port ACLs

Port ACLs are ACLs that are applied to Layer 2 interfaces on a switch. Port ACLs are supported only on physical interfaces and not on EtherChannel interfaces. Port ACLs can be applied to the interface in outbound and inbound direction. The following access lists are supported:

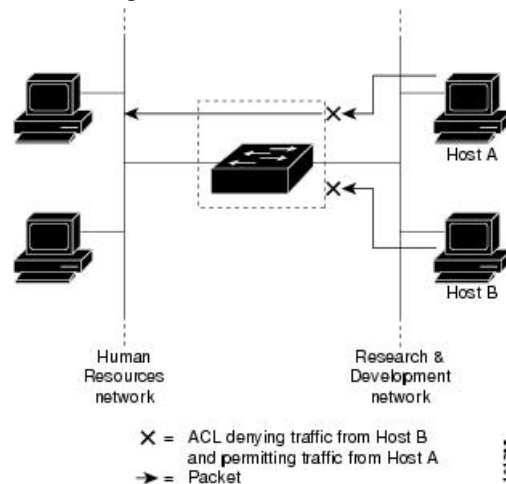
- Standard IP access lists using source addresses
- Extended IP access lists using source and destination addresses and optional protocol type information

- MAC extended access lists using source and destination MAC addresses and optional protocol type information

The switch examines ACLs on an interface and permits or denies packet forwarding based on how the packet matches the entries in the ACL. In this way, ACLs control access to a network or to part of a network.

Figure 94: Using ACLs to Control Traffic in a Network

This is an example of using port ACLs to control access to a network when all workstations are in the same VLAN. ACLs applied at the Layer 2 input would allow Host A to access the Human Resources network, but prevent Host B from accessing the same network. Port ACLs can only be applied to Layer 2 interfaces in the



inbound direction.

When you apply a port ACL to a trunk port, the ACL filters traffic on all VLANs present on the trunk port. When you apply a port ACL to a port with voice VLAN, the ACL filters traffic on both data and voice VLANs.

With port ACLs, you can filter IP traffic by using IP access lists and non-IP traffic by using MAC addresses. You can filter both IP and non-IP traffic on the same Layer 2 interface by applying both an IP access list and a MAC access list to the interface.



Note You cannot apply more than one IP access list and one MAC access list to a Layer 2 interface. If an IP access list or MAC access list is already configured on a Layer 2 interface and you apply a new IP access list or MAC access list to the interface, the new ACL replaces the previously configured one.

Router ACLs

You can apply router ACLs on switch virtual interfaces (SVIs), which are Layer 3 interfaces to VLANs; on physical Layer 3 interfaces; and on Layer 3 EtherChannel interfaces. You apply router ACLs on interfaces for specific directions (inbound or outbound). You can apply one router ACL in each direction on an interface.

The switch supports these access lists for IPv4 traffic:

- Standard IP access lists use source addresses for matching operations.
- Extended IP access lists use source and destination addresses and optional protocol type information for matching operations.

As with port ACLs, the switch examines ACLs associated with features configured on a given interface. As packets enter the switch on an interface, ACLs associated with all inbound features configured on that interface are examined. After packets are routed and before they are forwarded to the next hop, all ACLs associated with outbound features configured on the egress interface are examined.

ACLs permit or deny packet forwarding based on how the packet matches the entries in the ACL, and can be used to control access to a network or to part of a network.

Access Control Entries

An ACL contains an ordered list of access control entries (ACEs). Each ACE specifies *permit* or *deny* and a set of conditions the packet must satisfy in order to match the ACE. The meaning of *permit* or *deny* depends on the context in which the ACL is used.

ACEs and Fragmented and Unfragmented Traffic

IP packets can be fragmented as they cross the network. When this happens, only the fragment containing the beginning of the packet contains the Layer 4 information, such as TCP or UDP port numbers, ICMP type and code, and so on. All other fragments are missing this information.

Some access control entries (ACEs) do not check Layer 4 information and therefore can be applied to all packet fragments. ACEs that do test Layer 4 information cannot be applied in the standard manner to most of the fragments in a fragmented IP packet. When the fragment contains no Layer 4 information and the ACE tests some Layer 4 information, the matching rules are modified:

- Permit ACEs that check the Layer 3 information in the fragment (including protocol type, such as TCP, UDP, and so on) are considered to match the fragment regardless of what the missing Layer 4 information might have been.



Note For TCP ACEs with L4 Ops, the fragmented packets will be dropped per RFC 1858.

- Deny ACEs that check Layer 4 information never match a fragment unless the fragment contains Layer 4 information.

ACEs and Fragmented and Unfragmented Traffic Examples

Consider access list 102, configured with these commands, applied to three fragmented packets:

```
Device(config)# access-list 102 permit tcp any host 10.1.1.1 eq smtp
Device(config)# access-list 102 deny tcp any host 10.1.1.2 eq telnet
Device(config)# access-list 102 permit tcp any host 10.1.1.2
Device(config)# access-list 102 deny tcp any any
```



Note In the first and second ACEs in the examples, the *eq* keyword after the destination address means to test for the TCP-destination-port well-known numbers equaling Simple Mail Transfer Protocol (SMTP) and Telnet, respectively.

- Packet A is a TCP packet from host 10.2.2.2., port 65000, going to host 10.1.1.1 on the SMTP port. If this packet is fragmented, the first fragment matches the first ACE (a permit) as if it were a complete packet because all Layer 4 information is present. The remaining fragments also match the first ACE, even though they do not contain the SMTP port information, because the first ACE only checks Layer 3 information when applied to fragments. The information in this example is that the packet is TCP and that the destination is 10.1.1.1.
- Packet B is from host 10.2.2.2, port 65001, going to host 10.1.1.2 on the Telnet port. If this packet is fragmented, the first fragment matches the second ACE (a deny) because all Layer 3 and Layer 4 information is present. The remaining fragments in the packet do not match the second ACE because they are missing Layer 4 information. Instead, they match the third ACE (a permit).

Because the first fragment was denied, host 10.1.1.2 cannot reassemble a complete packet, so packet B is effectively denied. However, the later fragments that are permitted will consume bandwidth on the network and resources of host 10.1.1.2 as it tries to reassemble the packet.

- Fragmented packet C is from host 10.2.2.2, port 65001, going to host 10.1.1.3, port ftp. If this packet is fragmented, the first fragment matches the fourth ACE (a deny). All other fragments also match the fourth ACE because that ACE does not check any Layer 4 information and because Layer 3 information in all fragments shows that they are being sent to host 10.1.1.3, and the earlier permit ACEs were checking different hosts.



CHAPTER 67

Configuring IPv4 Access Control Lists

Access control lists (ACLs) perform packet filtering to control which packets move through the network and where. Such control provides security by helping to limit network traffic, restrict the access of users and devices to the network, and prevent traffic from leaving a network. IP access lists can reduce the chance of spoofing and denial-of-service attacks and allow dynamic, temporary user access through a firewall.

IP access lists can also be used for purposes other than security, such as bandwidth control, restricting the content of routing updates, redistributing routes, triggering dial-on-demand (DDR) calls, limiting debug output, and identifying or classifying traffic for quality of service (QoS) features. This module provides an overview of IP access lists.

- [Prerequisites for Configuring IPv4 Access Control Lists, on page 1307](#)
- [Restrictions for Configuring IPv4 Access Control Lists, on page 1307](#)
- [Information About Configuring IPv4 Access Control Lists, on page 1308](#)
- [How to Configure ACLs, on page 1316](#)
- [Monitoring IPv4 ACLs, on page 1336](#)
- [Configuration Examples for ACLs, on page 1337](#)
- [Examples: Troubleshooting ACLs, on page 1345](#)
- [Additional References, on page 1346](#)
- [Feature Information for IPv4 Access Control Lists, on page 1347](#)

Prerequisites for Configuring IPv4 Access Control Lists

This section lists the prerequisites for configuring network security with access control lists (ACLs).

- On switches running the LAN base feature set, VLAN maps are supported.

Restrictions for Configuring IPv4 Access Control Lists

General Network Security

The following are restrictions for configuring network security with ACLs:

- Not all commands that accept a numbered ACL accept a named ACL. ACLs for packet filters and route filters on interfaces can use a name.

- A standard ACL and an extended ACL cannot have the same name.
- Though visible in the command-line help strings, **AppleTalk** is not supported as a matching condition for the **deny** and **permit** MAC access-list configuration mode commands.
- ACL wild card is not supported in downstream client policy.

IPv4 ACL Network Interfaces

The following restrictions apply to IPv4 ACLs to network interfaces:

- When controlling access to an interface, you can use a named or numbered ACL.
- If you apply an ACL to a Layer 3 interface and routing is not enabled on the switch, the ACL only filters packets that are intended for the CPU, such as SNMP, Telnet, or web traffic.
- You do not have to enable routing to apply ACLs to Layer 2 interfaces.

MAC ACLs on a Layer 2 Interface

After you create a MAC ACL, you can apply it to a Layer 2 interface to filter non-IP traffic coming in that interface. When you apply the MAC ACL, consider these guidelines:

- You can apply no more than one IP access list and one MAC access list to the same Layer 2 interface. The IP access list filters only IP packets, and the MAC access list filters non-IP packets.
- A Layer 2 interface can have only one MAC access list. If you apply a MAC access list to a Layer 2 interface that has a MAC ACL configured, the new ACL replaces the previously configured one.



Note The **mac access-group** interface configuration command is only valid when applied to a physical Layer 2 interface. You cannot use the command on EtherChannel port channels.

IP Access List Entry Sequence Numbering

- This feature does not support dynamic, reflexive, or firewall access lists.

Information About Configuring IPv4 Access Control Lists

ACL Overview

Packet filtering can help limit network traffic and restrict network use by certain users or devices. ACLs filter traffic as it passes through a router or switch and permit or deny packets crossing specified interfaces or VLANs. An ACL is a sequential collection of permit and deny conditions that apply to packets. When a packet is received on an interface, the switch compares the fields in the packet against any applied ACLs to verify that the packet has the required permissions to be forwarded, based on the criteria specified in the access lists. One by one, it tests packets against the conditions in an access list. The first match decides whether the switch accepts or rejects the packets. Because the switch stops testing after the first match, the order of conditions in the list is critical. If no conditions match, the switch rejects the packet. If there are no restrictions, the switch

forwards the packet; otherwise, the switch drops the packet. The switch can use ACLs on all packets it forwards, including packets bridged within a VLAN.

You configure access lists on a router or Layer 3 switch to provide basic security for your network. If you do not configure ACLs, all packets passing through the switch could be allowed onto all parts of the network. You can use ACLs to control which hosts can access different parts of a network or to decide which types of traffic are forwarded or blocked at router interfaces. For example, you can allow e-mail traffic to be forwarded but not Telnet traffic. ACLs can be configured to block inbound traffic, outbound traffic, or both.

Standard and Extended IPv4 ACLs

This section describes IP ACLs.

An ACL is a sequential collection of permit and deny conditions. One by one, the switch tests packets against the conditions in an access list. The first match determines whether the switch accepts or rejects the packet. Because the switch stops testing after the first match, the order of the conditions is critical. If no conditions match, the switch denies the packet.

The software supports these types of ACLs or access lists for IPv4:

- Standard IP access lists use source addresses for matching operations.
- Extended IP access lists use source and destination addresses for matching operations and optional protocol-type information for finer granularity of control.

IPv4 ACL Switch Unsupported Features

Configuring IPv4 ACLs on the switch is the same as configuring IPv4 ACLs on other Cisco switches and routers.

The following ACL-related features are not supported:

- Non-IP protocol ACLs
- IP accounting
- Reflexive ACLs and dynamic ACLs are not supported.

Access List Numbers

The number you use to denote your ACL shows the type of access list that you are creating.

This lists the access-list number and corresponding access list type and shows whether or not they are supported in the switch. The switch supports IPv4 standard and extended access lists, numbers 1 to 199 and 1300 to 2699.

Table 133: Access List Numbers

Access List Number	Type	Supported
1–99	IP standard access list	Yes
100–199	IP extended access list	Yes
200–299	Protocol type-code access list	No

Access List Number	Type	Supported
300–399	DECnet access list	No
400–499	XNS standard access list	No
500–599	XNS extended access list	No
600–699	AppleTalk access list	No
700–799	48-bit MAC address access list	No
800–899	IPX standard access list	No
900–999	IPX extended access list	No
1000–1099	IPX SAP access list	No
1100–1199	Extended 48-bit MAC address access list	No
1200–1299	IPX summary address access list	No
1300–1999	IP standard access list (expanded range)	Yes
2000–2699	IP extended access list (expanded range)	Yes

In addition to numbered standard and extended ACLs, you can also create standard and extended named IP ACLs by using the supported numbers. That is, the name of a standard IP ACL can be 1 to 99; the name of an extended IP ACL can be 100 to 199. The advantage of using named ACLs instead of numbered lists is that you can delete individual entries from a named list.

Numbered Standard IPv4 ACLs

When creating an ACL, remember that, by default, the end of the ACL contains an implicit deny statement for all packets that it did not find a match for before reaching the end. With standard access lists, if you omit the mask from an associated IP host address ACL specification, 0.0.0.0 is assumed to be the mask.

The switch always rewrites the order of standard access lists so that entries with **host** matches and entries with matches having a *don't care* mask of 0.0.0.0 are moved to the top of the list, above any entries with non-zero *don't care* masks. Therefore, in **show** command output and in the configuration file, the ACEs do not necessarily appear in the order in which they were entered.

After creating a numbered standard IPv4 ACL, you can apply it to VLANs, to terminal lines, or to interfaces.

Numbered Extended IPv4 ACLs

Although standard ACLs use only source addresses for matching, you can use extended ACL source and destination addresses for matching operations and optional protocol type information for finer granularity of control. When you are creating ACEs in numbered extended access lists, remember that after you create the ACL, any additions are placed at the end of the list. You cannot reorder the list or selectively add or remove ACEs from a numbered list.

The switch does not support dynamic or reflexive access lists. It also does not support filtering based on the type of service (ToS) minimize-monetary-cost bit.

Some protocols also have specific parameters and keywords that apply to that protocol.

You can define an extended TCP, UDP, ICMP, IGMP, or other IP ACL. The switch also supports these IP protocols:



Note ICMP echo-reply cannot be filtered. All other ICMP codes or types can be filtered.

These IP protocols are supported:

- Authentication Header Protocol (**ahp**)
- Encapsulation Security Payload (**esp**)
- Enhanced Interior Gateway Routing Protocol (**eigrp**)
- generic routing encapsulation (**gre**)
- Internet Control Message Protocol (**icmp**)
- Internet Group Management Protocol (**igmp**)
- any Interior Protocol (**ip**)
- IP in IP tunneling (**ipinip**)
- KA9Q NOS-compatible IP over IP tunneling (**nos**)
- Open Shortest Path First routing (**ospf**)
- Payload Compression Protocol (**pcp**)
- Protocol-Independent Multicast (**pim**)
- Transmission Control Protocol (**tcp**)
- User Datagram Protocol (**udp**)

Named IPv4 ACLs

You can identify IPv4 ACLs with an alphanumeric string (a name) rather than a number. You can use named ACLs to configure more IPv4 access lists in a router than if you were to use numbered access lists. If you identify your access list with a name rather than a number, the mode and command syntax are slightly different. However, not all commands that use IP access lists accept a named access list.



Note The name you give to a standard or extended ACL can also be a number in the supported range of access list numbers. That is, the name of a standard IP ACL can be 1 to 99 and . The advantage of using named ACLs instead of numbered lists is that you can delete individual entries from a named list.

Consider these guidelines before configuring named ACLs:

- Numbered ACLs are also available.

- A standard ACL and an extended ACL cannot have the same name.
- You can use standard or extended ACLs (named or numbered) in VLAN maps.

Benefits of Using the Named ACL Support for Noncontiguous Ports on an Access Control Entry Feature

The Named ACL Support for Noncontiguous Ports on an Access Control Entry feature allows you to specify noncontiguous ports in a single access control entry, which greatly reduces the number of entries required in an access control list when several entries have the same source address, destination address, and protocol, but differ only in the ports.

This feature greatly reduces the number of access control entries (ACEs) required in an access control list to handle multiple entries for the same source address, destination address, and protocol. If you maintain large numbers of ACEs, use this feature to consolidate existing groups of access list entries wherever it is possible and when you create new access list entries. When you configure access list entries with noncontiguous ports, you will have fewer access list entries to maintain.

Benefits of IP Access List Entry Sequence Numbering

The ability to apply sequence numbers to IP access list entries simplifies access list changes. Prior to the IP Access List Entry Sequence Numbering feature, there was no way to specify the position of an entry within an access list. If a user wanted to insert an entry (statement) in the middle of an existing list, all of the entries after the desired position had to be removed, then the new entry was added, and then all the removed entries had to be reentered. This method was cumbersome and error prone.

This feature allows users to add sequence numbers to access list entries and resequence them. When a user adds a new entry, the user chooses the sequence number so that it is in a desired position in the access list. If necessary, entries currently in the access list can be resequenced to create room to insert the new entry.

Sequence Numbering Behavior

- For backward compatibility with previous releases, if entries with no sequence numbers are applied, the first entry is assigned a sequence number of 10, and successive entries are incremented by 10. The maximum sequence number is 2147483647. If the generated sequence number exceeds this maximum number, the following message is displayed:

```
Exceeded maximum sequence number.
```

- If the user enters an entry without a sequence number, it is assigned a sequence number that is 10 greater than the last sequence number in that access list and is placed at the end of the list.
- If the user enters an entry that matches an already existing entry (except for the sequence number), then no changes are made.
- If the user enters a sequence number that is already present, the following error message is generated:

```
Duplicate sequence number.
```

- If a new access list is entered from global configuration mode, then sequence numbers for that access list are generated automatically.
- Distributed support is provided so that the sequence numbers of entries in the Route Processor (RP) and line card are in synchronization at all times.

- Sequence numbers are not nvgened. That is, the sequence numbers themselves are not saved. In the event that the system is reloaded, the configured sequence numbers revert to the default sequence starting number and increment. The function is provided for backward compatibility with software releases that do not support sequence numbering.
- This feature works with named and numbered, standard and extended IP access lists.

Including comments in ACLs

You can use the **remark** keyword to include comments (remarks) about entries in any IP standard or extended ACL. The remarks make the ACL easier for you to understand and scan. Each remark line is limited to 100 characters.

The remark can go before or after a permit or deny statement. You should be consistent about where you put the remark so that it is clear which remark describes which permit or deny statement. For example, it would be confusing to have some remarks before the associated permit or deny statements and some remarks after the associated statements.

To include a comment for IP numbered standard or extended ACLs, use the **access-list** *access-list number* **remark** *remark* global configuration command. To remove the remark, use the **no** form of this command.

The following is an example of a remark that describes function of the subsequent deny statement:

```
ip access-list extended telnetting
  remark Do not allow host1 subnet to telnet out
deny tcp host 172.16.2.88 any eq telnet
```

Hardware and Software Treatment of IP ACLs

ACL processing is performed in hardware. If the hardware reaches its capacity to store ACL configurations, all packets on that interface are dropped.



Note If an ACL configuration cannot be implemented in hardware due to an out-of-resource condition on a switch or stack member, then only the traffic in that VLAN arriving on that switch is affected.

When you enter the **show ip access-lists** privileged EXEC command, the match count displayed does not account for packets that are access controlled in hardware. Use the **show ip access-lists hardware** privileged EXEC command to obtain some basic hardware ACL statistics for switched and routed packets.

Time Ranges for ACLs

You can selectively apply extended ACLs based on the time of day and the week by using the **time-range** global configuration command. First, define a time-range name and set the times and the dates or the days of the week in the time range. Then enter the time-range name when applying an ACL to set restrictions to the access list. You can use the time range to define when the permit or deny statements in the ACL are in effect, for example, during a specified time period or on specified days of the week. The **time-range** keyword and argument are referenced in the named and numbered extended ACL task tables.

These are some benefits of using time ranges:

- You have more control over permitting or denying a user access to resources, such as an application (identified by an IP address/mask pair and a port number).
- You can control logging messages. ACL entries can be set to log traffic only at certain times of the day. Therefore, you can simply deny access without needing to analyze many logs generated during peak hours.

Time-based access lists trigger CPU activity because the new configuration of the access list must be merged with other features and the combined configuration loaded into the hardware memory. For this reason, you should be careful not to have several access lists configured to take affect in close succession (within a small number of minutes of each other.)



Note The time range relies on the switch system clock; therefore, you need a reliable clock source. We recommend that you use Network Time Protocol (NTP) to synchronize the switch clock.

IPv4 ACL Interface Considerations

When you apply the **ip access-group** interface configuration command to a Layer 3 interface (an SVI, a Layer 3 EtherChannel, or a routed port), the interface must have been configured with an IP address. Layer 3 access groups filter packets that are routed or are received by Layer 3 processes on the CPU. They do not affect packets bridged within a VLAN.

For inbound ACLs, after receiving a packet, the switch checks the packet against the ACL. If the ACL permits the packet, the switch continues to process the packet. If the ACL rejects the packet, the switch discards the packet.

For outbound ACLs, after receiving and routing a packet to a controlled interface, the switch checks the packet against the ACL. If the ACL permits the packet, the switch sends the packet. If the ACL rejects the packet, the switch discards the packet.

By default, the input interface sends ICMP Unreachable messages whenever a packet is discarded, regardless of whether the packet was discarded because of an ACL on the input interface or because of an ACL on the output interface. ICMP Unreachables are normally limited to no more than one every one-half second per input interface, but this can be changed by using the **ip icmp rate-limit unreachable** global configuration command.

When you apply an undefined ACL to an interface, the switch acts as if the ACL has not been applied to the interface and permits all packets. Remember this behavior if you use undefined ACLs for network security.

Apply an Access Control List to an Interface

With some protocols, you can apply up to two access lists to an interface: one inbound access list and one outbound access list. With other protocols, you apply only one access list that checks both inbound and outbound packets.

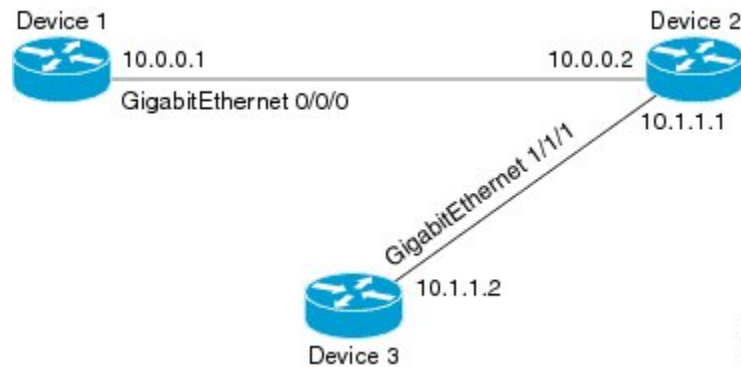
If the access list is inbound, when a device receives a packet, Cisco software checks the access list's criteria statements for a match. If the packet is permitted, the software continues to process the packet. If the packet is denied, the software discards the packet.

If the access list is outbound, after receiving and routing a packet to the outbound interface, Cisco software checks the access list's criteria statements for a match. If the packet is permitted, the software transmits the packet. If the packet is denied, the software discards the packet.



Note Access lists that are applied to interfaces on a device do not filter traffic that originates from that device.

Figure 95: Topology for Applying Access Control Lists



The figure above shows that Device 2 is a bypass device that is connected to Device 1 and Device 3. An outbound access list is applied to Gigabit Ethernet interface 0/0/0 on Device 1. When you ping Device 3 from Device 1, the access list does not check for packets going outbound because the traffic is locally generated.

The access list check is bypassed for locally generated packets, which are always outbound.

By default, an access list that is applied to an outbound interface for matching locally generated traffic will bypass the outbound access list check; but transit traffic is subjected to the outbound access list check.



Note The behavior described above applies to all single-CPU platforms that run Cisco software.

ACL Logging

The switch software can provide logging messages about packets permitted or denied by a standard IP access list. That is, any packet that matches the ACL causes an informational logging message about the packet to be sent to the console. The level of messages logged to the console is controlled by the **logging console** commands controlling the syslog messages.



Note ACL logging is only supported for RACL.



Note Because routing is done in hardware and logging is done in software, if a large number of packets match a *permit* or *deny* ACE containing a **log** keyword, the software might not be able to match the hardware processing rate, and not all packets will be logged.

The first packet that triggers the ACL causes a logging message right away, and subsequent packets are collected over 5-minute intervals before they appear or logged. The logging message includes the access list

number, whether the packet was permitted or denied, the source IP address of the packet, and the number of packets from that source permitted or denied in the prior 5-minute interval.



Note The logging facility might drop some logging message packets if there are too many to be handled or if there is more than one logging message to be handled in 1 second. This behavior prevents the router from crashing due to too many logging packets. Therefore, the logging facility should not be used as a billing tool or an accurate source of the number of matches to an access list.

How to Configure ACLs

Configuring IPv4 ACLs

Follow the procedure given below to use IP ACLs on the switch:

SUMMARY STEPS

1. Create an ACL by specifying an access list number or name and the access conditions.
2. Apply the ACL to interfaces or terminal lines. You can also apply standard and extended IP ACLs to VLAN maps.

DETAILED STEPS

-
- Step 1** Create an ACL by specifying an access list number or name and the access conditions.
- Step 2** Apply the ACL to interfaces or terminal lines. You can also apply standard and extended IP ACLs to VLAN maps.
-

Creating a Numbered Standard ACL

Follow these steps to create a numbered standard ACL:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **access-list** *access-list-number* {deny | permit} *source source-wildcard*]
4. **end**
5. **show running-config**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	access-list <i>access-list-number</i> { deny permit } <i>source source-wildcard</i>] Example: Device(config)# access-list 2 deny your_host	<p>Defines a standard IPv4 access list by using a source address and wildcard.</p> <p>The <i>access-list-number</i> is a decimal number from 1 to 99 or 1300 to 1999.</p> <p>Enter deny or permit to specify whether to deny or permit access if conditions are matched.</p> <p>The <i>source</i> is the source address of the network or host from which the packet is being sent specified as:</p> <ul style="list-style-type: none"> • The 32-bit quantity in dotted-decimal format. • The keyword any as an abbreviation for <i>source</i> and <i>source-wildcard</i> of 0.0.0.0 255.255.255.255. You do not need to enter a source-wildcard. • The keyword host as an abbreviation for source and <i>source-wildcard</i> of <i>source</i> 0.0.0.0. <p>(Optional) The <i>source-wildcard</i> applies wildcard bits to the source.</p> <p>Note Logging is supported only on ACLs attached to Layer 3 interfaces.</p>
Step 4	end Example: Device(config)# end	Returns to privileged EXEC mode.
Step 5	show running-config Example: Device# show running-config	Verifies your entries.

	Command or Action	Purpose
Step 6	copy running-config startup-config Example: Device# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Creating a Numbered Extended ACL

Follow these steps to create a numbered extended ACL:

SUMMARY STEPS

- configure terminal**
- access-list** *access-list-number* {deny | permit} *protocol source source-wildcard destination destination-wildcard* [precedence *precedence*] [tos *tos*] [fragments] [log [log-input] [time-range *time-range-name*] [dscp *dscp*]
- access-list** *access-list-number* {deny | permit} **tcp** *source source-wildcard* [*operator port*] *destination destination-wildcard* [*operator port*] [established] [precedence *precedence*] [tos *tos*] [fragments] [log [log-input] [time-range *time-range-name*] [dscp *dscp*] [flag]
- access-list** *access-list-number* {deny | permit} **udp** *source source-wildcard* [*operator port*] *destination destination-wildcard* [*operator port*] [precedence *precedence*] [tos *tos*] [fragments] [log [log-input] [time-range *time-range-name*] [dscp *dscp*]
- access-list** *access-list-number* {deny | permit} **icmp** *source source-wildcard destination destination-wildcard* [*icmp-type* | [[*icmp-type icmp-code*] | [*icmp-message*]]] [precedence *precedence*] [tos *tos*] [fragments] [log [log-input] [time-range *time-range-name*] [dscp *dscp*]
- access-list** *access-list-number* {deny | permit} **igmp** *source source-wildcard destination destination-wildcard* [*igmp-type*] [precedence *precedence*] [tos *tos*] [fragments] [log [log-input] [time-range *time-range-name*] [dscp *dscp*]
- end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode.
Step 2	access-list <i>access-list-number</i> {deny permit} <i>protocol source source-wildcard destination destination-wildcard</i> [precedence <i>precedence</i>] [tos <i>tos</i>] [fragments] [log [log-input] [time-range <i>time-range-name</i>] [dscp <i>dscp</i>] Example: Device(config)# <code>access-list 101 permit ip host</code>	Defines an extended IPv4 access list and the access conditions. The <i>access-list-number</i> is a decimal number from 100 to 199 or 2000 to 2699. Enter deny or permit to specify whether to deny or permit the packet if conditions are matched.

	Command or Action	Purpose
	10.1.1.2 any precedence 0 tos 0 log	<p>For <i>protocol</i>, enter the name or number of an P protocol: ahp, eigrp, esp, gre, icmp, igmp, igrp, ip, ipinip, nos, ospf, pcp, pim, tcp, or udp, or an integer in the range 0 to 255 representing an IP protocol number. To match any Internet protocol (including ICMP, TCP, and UDP), use the keyword ip.</p> <p>Note This step includes options for most IP protocols. For additional specific parameters for TCP, UDP, ICMP, and IGMP, see the following steps.</p> <p>The <i>source</i> is the number of the network or host from which the packet is sent.</p> <p>The <i>source-wildcard</i> applies wildcard bits to the source.</p> <p>The <i>destination</i> is the network or host number to which the packet is sent.</p> <p>The <i>destination-wildcard</i> applies wildcard bits to the destination.</p> <p>Source, source-wildcard, destination, and destination-wildcard can be specified as:</p> <ul style="list-style-type: none"> • The 32-bit quantity in dotted-decimal format. • The keyword any for 0.0.0.0 255.255.255.255 (any host). • The keyword host for a single host 0.0.0.0. <p>The other keywords are optional and have these meanings:</p> <ul style="list-style-type: none"> • precedence—Enter to match packets with a precedence level specified as a number from 0 to 7 or by name: routine (0), priority (1), immediate (2), flash (3), flash-override (4), critical (5), internet (6), network (7). • fragments—Enter to check non-initial fragments. • tos—Enter to match by type of service level, specified by a number from 0 to 15 or a name: normal (0), max-reliability (2), max-throughput (4), min-delay (8). • log—Enter to create an informational logging message to be sent to the console about the packet that matches the entry or log-input to include the input interface in the log entry. • time-range—Specify the time-range name.

	Command or Action	Purpose
		<ul style="list-style-type: none"> • dscp—Enter to match packets with the DSCP value specified by a number from 0 to 63, or use the question mark (?) to see a list of available values. <p>Note If you enter a dscp value, you cannot enter tos or precedence. You can enter both a tos and a precedence value with no dscp.</p>
Step 3	<p>access-list <i>access-list-number</i> {deny permit} tcp <i>source source-wildcard</i> [<i>operator port</i>] <i>destination destination-wildcard</i> [<i>operator port</i>] [established] [precedence <i>precedence</i>] [tos <i>tos</i>] [fragments] [log [log-input] [time-range <i>time-range-name</i>] [dscp <i>dscp</i>] [<i>flag</i>]</p> <p>Example:</p> <pre>Device(config)# access-list 101 permit tcp any any eq 500</pre>	<p>Defines an extended TCP access list and the access conditions.</p> <p>The parameters are the same as those described for an extended IPv4 ACL, with these exceptions:</p> <p>(Optional) Enter an <i>operator</i> and <i>port</i> to compare source (if positioned after <i>source source-wildcard</i>) or destination (if positioned after <i>destination destination-wildcard</i>) port. Possible operators include eq (equal), gt (greater than), lt (less than), neq (not equal), and range (inclusive range). Operators require a port number (range requires two port numbers separated by a space).</p> <p>Enter the <i>port</i> number as a decimal number (from 0 to 65535) or the name of a TCP port. Use only TCP port numbers or names when filtering TCP.</p> <p>The other optional keywords have these meanings:</p> <ul style="list-style-type: none"> • established—Enter to match an established connection. This has the same function as matching on the ack or rst flag. • <i>flag</i>—Enter one of these flags to match by the specified TCP header bits: ack (acknowledge), fin (finish), psh (push), rst (reset), syn (synchronize), or urg (urgent).
Step 4	<p>access-list <i>access-list-number</i> {deny permit} udp <i>source source-wildcard</i> [<i>operator port</i>] <i>destination destination-wildcard</i> [<i>operator port</i>] [precedence <i>precedence</i>] [tos <i>tos</i>] [fragments] [log [log-input] [time-range <i>time-range-name</i>] [dscp <i>dscp</i>]</p> <p>Example:</p> <pre>Device(config)# access-list 101 permit udp any any eq 100</pre>	<p>(Optional) Defines an extended UDP access list and the access conditions.</p> <p>The UDP parameters are the same as those described for TCP except that the [<i>operator</i> [<i>port</i>]] port number or name must be a UDP port number or name, and the flag and established keywords are not valid for UDP.</p>
Step 5	<p>access-list <i>access-list-number</i> {deny permit} icmp <i>source source-wildcard</i> <i>destination destination-wildcard</i> [<i>icmp-type</i> [[<i>icmp-type icmp-code</i>] [<i>icmp-message</i>]]] [precedence <i>precedence</i>] [tos <i>tos</i>] [fragments] [log [log-input] [time-range <i>time-range-name</i>] [dscp <i>dscp</i>]</p>	<p>Defines an extended ICMP access list and the access conditions.</p> <p>The ICMP parameters are the same as those described for most IP protocols in an extended IPv4 ACL, with the</p>

	Command or Action	Purpose
	<p>Example:</p> <pre>Device(config)# access-list 101 permit icmp any any 200</pre>	<p>addition of the ICMP message type and code parameters. These optional keywords have these meanings:</p> <ul style="list-style-type: none"> • <i>icmp-type</i>—Enter to filter by ICMP message type, a number from 0 to 255. • <i>icmp-code</i>—Enter to filter ICMP packets that are filtered by the ICMP message code type, a number from 0 to 255. • <i>icmp-message</i>—Enter to filter ICMP packets by the ICMP message type name or the ICMP message type and code name.
Step 6	<p>access-list <i>access-list-number</i> {deny permit} igmp <i>source source-wildcard destination destination-wildcard</i> [<i>igmp-type</i>] [precedence <i>precedence</i>] [tos <i>tos</i>] [fragments] [log [log-input] [time-range <i>time-range-name</i>] [dscp <i>dscp</i>]</p> <p>Example:</p> <pre>Device(config)# access-list 101 permit igmp any any 14</pre>	<p>(Optional) Defines an extended IGMP access list and the access conditions.</p> <p>The IGMP parameters are the same as those described for most IP protocols in an extended IPv4 ACL, with this optional parameter.</p> <p><i>igmp-type</i>—To match IGMP message type, enter a number from 0 to 15, or enter the message name: dvmrp, host-query, host-report, pim, or trace.</p>
Step 7	<p>end</p> <p>Example:</p> <pre>Device(config)# end</pre>	<p>Returns to privileged EXEC mode.</p>

Creating Named Standard ACLs

Follow these steps to create a standard ACL using names:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip access-list standard** *name*
4. Use one of the following:
 - **deny** {*source* [*source-wildcard*] | **host** *source* | **any**} [**log**]
 - **permit** {*source* [*source-wildcard*] | **host** *source* | **any**} [**log**]
5. **end**
6. **show running-config**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Device> enable</pre>	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	configure terminal Example: <pre>Device# configure terminal</pre>	Enters global configuration mode.
Step 3	ip access-list standard <i>name</i> Example: <pre>Device(config)# ip access-list standard 20</pre>	Defines a standard IPv4 access list using a name, and enter access-list configuration mode. The name can be a number from 1 to 99.
Step 4	Use one of the following: <ul style="list-style-type: none"> • deny {<i>source</i> [<i>source-wildcard</i>] host <i>source</i> any} [log] • permit {<i>source</i> [<i>source-wildcard</i>] host <i>source</i> any} [log] Example: <pre>Device(config-std-nacl)# deny 192.168.0.0 0.0.255.255 255.255.0.0 0.0.255.255</pre> or <pre>Device(config-std-nacl)# permit 10.108.0.0 0.0.0.0 255.255.255.0 0.0.0.0</pre>	In access-list configuration mode, specify one or more conditions denied or permitted to decide if the packet is forwarded or dropped. <ul style="list-style-type: none"> • host <i>source</i>—A source and source wildcard of <i>source</i> 0.0.0.0. • any—A source and source wildcard of 0.0.0.0 255.255.255.255.
Step 5	end Example: <pre>Device(config-std-nacl)# end</pre>	Returns to privileged EXEC mode.
Step 6	show running-config Example: <pre>Device# show running-config</pre>	Verifies your entries.

	Command or Action	Purpose
Step 7	copy running-config startup-config Example: Device# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Creating Extended Named ACLs

Follow the procedure given below to create an extended ACL using names:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip access-list extended *name***
4. **{deny | permit} protocol {source [source-wildcard] | host source | any} {destination [destination-wildcard] | host destination | any} [precedence precedence] [tos tos] [established] [log] [time-range time-range-name]**
5. **end**
6. **show running-config**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> <code>enable</code>	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode.
Step 3	ip access-list extended <i>name</i> Example: Device(config)# <code>ip access-list extended 150</code>	Defines an extended IPv4 access list using a name, and enter access-list configuration mode. The name can be a number from 100 to 199.
Step 4	{deny permit} protocol {source [source-wildcard] host source any} {destination [destination-wildcard] host destination any} [precedence precedence] [tos tos] [established] [log] [time-range time-range-name]	In access-list configuration mode, specify the conditions allowed or denied. Use the log keyword to get access list logging messages, including violations.

	Command or Action	Purpose
	<p>Example:</p> <pre>Device(config-ext-nacl)# permit 0 any any</pre>	<ul style="list-style-type: none"> • host source—A source and source wildcard of <i>source</i> 0.0.0.0. • host destination—A destination and destination wildcard of <i>destination</i> 0.0.0.0. • any—A source and source wildcard or destination and destination wildcard of 0.0.0.0 255.255.255.255.
Step 5	<p>end</p> <p>Example:</p> <pre>Device(config-ext-nacl)# end</pre>	Returns to privileged EXEC mode.
Step 6	<p>show running-config</p> <p>Example:</p> <pre>Device# show running-config</pre>	Verifies your entries.
Step 7	<p>copy running-config startup-config</p> <p>Example:</p> <pre>Device# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

When you are creating extended ACLs, remember that, by default, the end of the ACL contains an implicit deny statement for everything if it did not find a match before reaching the end. For standard ACLs, if you omit the mask from an associated IP host address access list specification, 0.0.0.0 is assumed to be the mask.

After you create an ACL, any additions are placed at the end of the list. You cannot selectively add ACL entries to a specific ACL. However, you can use **no permit** and **no deny** access-list configuration mode commands to remove entries from a named ACL.

Being able to selectively remove lines from a named ACL is one reason you might use named ACLs instead of numbered ACLs.

What to do next

After creating a named ACL, you can apply it to interfaces or to VLANs.

Configuring an Access Control Entry with Noncontiguous Ports

Perform this task to create access list entries that use noncontiguous TCP or UDP port numbers. Although this task uses TCP ports, you could use the UDP syntax of the **permit** and **deny** commands to filter noncontiguous UDP ports.

Although this task uses a **permit** command first, use the **permit** and **deny** commands in the order that achieves your filtering goals.



Note The ACL—Named ACL Support for Noncontiguous Ports on an Access Control Entry feature can be used only with named, extended ACLs.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip access-list extended** *access-list-name*
4. [*sequence-number*] **permit tcp** *source source-wildcard* [*operator port [port]*] *destination destination-wildcard* [*operator [port]*] [**established** {**match-any** | **match-all**} {+ | -} *flag-name*] [**precedence** *precedence*] [**tos** *tos*] [**log**] [**time-range** *time-range-name*] [**fragments**]
5. [*sequence-number*] **deny tcp** *source source-wildcard* [*operator port [port]*] *destination destination-wildcard* [*operator [port]*] [**established** {**match-any** | **match-all**} {+ | -} *flag-name*] [**precedence** *precedence*] [**tos** *tos*] [**log**] [**time-range** *time-range-name*] [**fragments**]
6. Repeat Step 4 or Step 5 as necessary, adding statements by sequence number where you planned. Use the **no** *sequence-number* command to delete an entry.
7. **end**
8. **show ip access-lists** *access-list-name*

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	ip access-list extended <i>access-list-name</i> Example: Device(config)# ip access-list extended acl-extd-1	Specifies the IP access list by name and enters named access list configuration mode.
Step 4	[<i>sequence-number</i>] permit tcp <i>source source-wildcard</i> [<i>operator port [port]</i>] <i>destination destination-wildcard</i> [<i>operator [port]</i>] [established { match-any match-all } {+ -} <i>flag-name</i>] [precedence <i>precedence</i>] [tos <i>tos</i>] [log] [time-range <i>time-range-name</i>] [fragments] Example: Device(config-ext-nacl)# permit tcp any eq telnet ftp any eq 450 679	Specifies a permit statement in named IP access list configuration mode. <ul style="list-style-type: none"> • Operators include lt (less than), gt (greater than), eq (equal), neq (not equal), and range (inclusive range). • If the operator is positioned after the source and source-wildcard arguments, it must match the source port. If the operator is positioned after the destination and destination-wildcard arguments, it must match the destination port.

	Command or Action	Purpose
		<ul style="list-style-type: none"> The range operator requires two port numbers. You can configure up to 10 ports after the eq and neq operators. All other operators require one port number. To filter UDP ports, use the UDP syntax of this command.
Step 5	<p>[<i>sequence-number</i>] deny tcp <i>source source-wildcard</i> [<i>operator port [port]</i>] <i>destination destination-wildcard</i> [<i>operator [port]</i>] [established {match-any match-all} {+ -} <i>flag-name</i>] [precedence <i>precedence</i>] [tos <i>tos</i>] [log] [time-range <i>time-range-name</i>] [fragments]</p> <p>Example:</p> <pre>Device(config-ext-nacl)# deny tcp any neq 45 565 632</pre>	<p>(Optional) Specifies a deny statement in named access list configuration mode.</p> <ul style="list-style-type: none"> Operators include lt (less than), gt (greater than), eq (equal), neq (not equal), and range (inclusive range). If the <i>operator</i> is positioned after the <i>source</i> and <i>source-wildcard</i> arguments, it must match the source port. If the <i>operator</i> is positioned after the <i>destination</i> and <i>destination-wildcard</i> arguments, it must match the destination port. The range operator requires two port numbers. You can configure up to 10 ports after the eq and neq operators. All other operators require one port number. To filter UDP ports, use the UDP syntax of this command.
Step 6	Repeat Step 4 or Step 5 as necessary, adding statements by sequence number where you planned. Use the no <i>sequence-number</i> command to delete an entry.	Allows you to revise the access list.
Step 7	<p>end</p> <p>Example:</p> <pre>Device(config-ext-nacl)# end</pre>	(Optional) Exits named access list configuration mode and returns to privileged EXEC mode.
Step 8	<p>show ip access-lists <i>access-list-name</i></p> <p>Example:</p> <pre>Device# show ip access-lists kmd1</pre>	(Optional) Displays the contents of the access list.

Consolidating Access List Entries with Noncontiguous Ports into One Access List Entry

Perform this task to consolidate a group of access list entries with noncontiguous ports into one access list entry.

Although this task uses TCP ports, you could use the UDP syntax of the **permit** and **deny** commands to filter noncontiguous UDP ports.

Although this task uses a **permit** command first, use the **permit** and **deny** commands in the order that achieves your filtering goals.

SUMMARY STEPS

1. **enable**
2. **show ip access-lists** *access-list-name*
3. **configure terminal**
4. **ip access-list extended** *access-list-name*
5. **no** [*sequence-number*] **permit** *protocol source source-wildcard destination destination-wildcard*[**option** *option-name*] [**precedence** *precedence*][**tos** *tos*] [**log**] [**time-range** *time-range-name*] [**fragments**]
6. [*sequence-number*] **permit** *protocol source source-wildcard*[*operator port*[*port*]] *destination destination-wildcard*[*operator port*[*port*]] [**option** *option-name*] [**precedence** *precedence*][**tos** *tos*] [**log**] [**time-range** *time-range-name*] [**fragments**]
7. Repeat Steps 5 and 6 as necessary, adding **permit** or **deny** statements to consolidate access list entries where possible. Use the **no** *sequence-number* command to delete an entry.
8. **end**
9. **show ip access-lists** *access-list-name*

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none">• Enter your password if prompted.
Step 2	show ip access-lists <i>access-list-name</i> Example: Device# show ip access-lists mylist1	(Optional) Displays the contents of the IP access list. <ul style="list-style-type: none">• Review the output to see if you can consolidate any access list entries.
Step 3	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 4	ip access-list extended <i>access-list-name</i> Example: Device(config)# ip access-list extended mylist1	Specifies the IP access list by name and enters named access list configuration mode.
Step 5	no [<i>sequence-number</i>] permit <i>protocol source source-wildcard destination destination-wildcard</i> [option <i>option-name</i>] [precedence <i>precedence</i>][tos <i>tos</i>] [log] [time-range <i>time-range-name</i>] [fragments] Example: Device(config-ext-nacl)# no 10	Removes the redundant access list entry that can be consolidated. <ul style="list-style-type: none">• Repeat this step to remove entries to be consolidated because only the port numbers differ.• After this step is repeated to remove the access list entries 20, 30, and 40, for example, those entries are removed because they will be consolidated into one permit statement.• If a <i>sequence-number</i> is specified, the rest of the command syntax is optional.

	Command or Action	Purpose
Step 6	<p>[<i>sequence-number</i>] permit <i>protocol source source-wildcard</i>[<i>operator port</i>[<i>port</i>]] <i>destination destination-wildcard</i>[<i>operator port</i>[<i>port</i>]] [option <i>option-name</i>] [precedence <i>precedence</i>][tos <i>tos</i>] [log] [time-range <i>time-range-name</i>] [fragments]</p> <p>Example:</p> <pre>Device(config-ext-nacl)# permit tcp any neq 45 565 632 any eq 23 45 34 43</pre>	<p>Specifies a permit statement in named access list configuration mode.</p> <ul style="list-style-type: none"> • In this instance, a group of access list entries with noncontiguous ports was consolidated into one permit statement. • You can configure up to 10 ports after the eq and neq operators.
Step 7	Repeat Steps 5 and 6 as necessary, adding permit or deny statements to consolidate access list entries where possible. Use the no <i>sequence-number</i> command to delete an entry.	Allows you to revise the access list.
Step 8	<p>end</p> <p>Example:</p> <pre>Device(config-std-nacl)# end</pre>	(Optional) Exits named access list configuration mode and returns to privileged EXEC mode.
Step 9	<p>show ip access-lists <i>access-list-name</i></p> <p>Example:</p> <pre>Device# show ip access-lists mylist1</pre>	(Optional) Displays the contents of the access list.

Sequencing Access-List Entries and Revising the Access List

This task shows how to assign sequence numbers to entries in a named IP access list and how to add or delete an entry to or from an access list. When completing this task, keep the following points in mind:

- Resequencing the access list entries is optional. The resequencing step in this task is shown as required because that is one purpose of this feature and this task demonstrates that functionality.
- In the following procedure, the **permit** command is shown in Step 5 and the **deny** command is shown in Step 6. However, that order can be reversed. Use the order that suits the need of your configuration.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip access-list resequence** *access-list-name starting-sequence-number increment*
4. **ip access-list** {**standard**|**extended**} *access-list-name*
5. Do one of the following:
 - *sequence-number* **permit** *source source-wildcard*
 - *sequence-number* **permit** *protocol source source-wildcard destination destination-wildcard* [**precedence** *precedence*][**tos** *tos*] [**log**] [**time-range** *time-range-name*] [**fragments**]
6. Do one of the following:
 - *sequence-number* **deny** *source source-wildcard*

- *sequence-number deny protocol source source-wildcard destination destination-wildcard [precedence precedence][tos tos] [log] [time-range time-range-name] [fragments]*

7. Do one of the following:

- *sequence-number permit source source-wildcard*
- *sequence-number permit protocol source source-wildcard destination destination-wildcard [precedence precedence][tos tos] [log] [time-range time-range-name] [fragments]*

8. Do one of the following:

- *sequence-number deny source source-wildcard*
- *sequence-number deny protocol source source-wildcard destination destination-wildcard [precedence precedence][tos tos] [log] [time-range time-range-name] [fragments]*

9. Repeat Step 5 and/or Step 6 to add sequence number statements, as applicable.

10. **end**

11. **show ip access-lists** *access-list-name*

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	ip access-list resequence <i>access-list-name starting-sequence-number increment</i> Example: Device(config)# ip access-list resequence kmd1 100 15	Resequences the specified IP access list using the starting sequence number and the increment of sequence numbers.
Step 4	ip access-list {standard extended} <i>access-list-name</i> Example: Device(config)# ip access-list standard kmd1	Specifies the IP access list by name and enters named access list configuration mode. <ul style="list-style-type: none"> • If you specify standard, make sure you subsequently specify permit and/or deny statements using the standard access list syntax. • If you specify extended, make sure you subsequently specify permit and/or deny statements using the extended access list syntax.
Step 5	Do one of the following:	Specifies a permit statement in named IP access list mode.

	Command or Action	Purpose
	<ul style="list-style-type: none"> • <i>sequence-number</i> permit <i>source source-wildcard</i> • <i>sequence-number</i> permit <i>protocol source source-wildcard destination destination-wildcard</i> [precedence <i>precedence</i>][tos <i>tos</i>] [log] [time-range <i>time-range-name</i>] [fragments] <p>Example:</p> <pre>Device(config-std-nacl)# 105 permit 10.5.5.5 0.0.0.0 255</pre>	<ul style="list-style-type: none"> • This access list happens to use a permit statement first, but a deny statement could appear first, depending on the order of statements you need. • As the prompt indicates, this access list was a standard access list. If you had specified extended in Step 4, the prompt for this step would be Device(config-ext-nacl) and you would use the extended permit command syntax.
Step 6	<p>Do one of the following:</p> <ul style="list-style-type: none"> • <i>sequence-number</i> deny <i>source source-wildcard</i> • <i>sequence-number</i> deny <i>protocol source source-wildcard destination destination-wildcard</i> [precedence <i>precedence</i>][tos <i>tos</i>] [log] [time-range <i>time-range-name</i>] [fragments] <p>Example:</p> <pre>Device(config-std-nacl)# 105 deny 10.6.6.7 0.0.0.0 255</pre>	<p>(Optional) Specifies a deny statement in named IP access list mode.</p> <ul style="list-style-type: none"> • This access list uses a permit statement first, but a deny statement could appear first, depending on the order of statements you need. • As the prompt indicates, this access list was a standard access list. If you had specified extended in Step 4, the prompt for this step would be Device(config-ext-nacl) and you would use the extended deny command syntax.
Step 7	<p>Do one of the following:</p> <ul style="list-style-type: none"> • <i>sequence-number</i> permit <i>source source-wildcard</i> • <i>sequence-number</i> permit <i>protocol source source-wildcard destination destination-wildcard</i> [precedence <i>precedence</i>][tos <i>tos</i>] [log] [time-range <i>time-range-name</i>] [fragments] <p>Example:</p> <pre>Device(config-ext-nacl)# 150 permit tcp any any log</pre>	<p>Specifies a permit statement in named IP access list mode.</p> <ul style="list-style-type: none"> • This access list happens to use a permit statement first, but a deny statement could appear first, depending on the order of statements you need. • See the permit (IP) command for additional command syntax to permit upper layer protocols (ICMP, IGMP, TCP, and UDP). • Use the no <i>sequence-number</i> command to delete an entry.
Step 8	<p>Do one of the following:</p> <ul style="list-style-type: none"> • <i>sequence-number</i> deny <i>source source-wildcard</i> • <i>sequence-number</i> deny <i>protocol source source-wildcard destination destination-wildcard</i> [precedence <i>precedence</i>][tos <i>tos</i>] [log] [time-range <i>time-range-name</i>] [fragments] <p>Example:</p> <pre>Device(config-ext-nacl)# 150 deny tcp any any log</pre>	<p>(Optional) Specifies a deny statement in named IP access list mode.</p> <ul style="list-style-type: none"> • This access list happens to use a permit statement first, but a deny statement could appear first, depending on the order of statements you need. • See the deny (IP) command for additional command syntax to permit upper layer protocols (ICMP, IGMP, TCP, and UDP). • Use the no <i>sequence-number</i> command to delete an entry.
Step 9	Repeat Step 5 and/or Step 6 to add sequence number statements, as applicable.	Allows you to revise the access list.

	Command or Action	Purpose
Step 10	end Example: Device(config-std-nacl)# end	(Optional) Exits the configuration mode and returns to privileged EXEC mode.
Step 11	show ip access-lists <i>access-list-name</i> Example: Device# show ip access-lists kmdl	(Optional) Displays the contents of the IP access list.

Examples

Review the output of the **show ip access-lists** command to see that the access list includes the new entries:

```
Device# show ip access-lists kmdl

Standard IP access list kmdl
100 permit 10.4.4.0, wildcard bits 0.0.0.255
105 permit 10.5.5.0, wildcard bits 0.0.0.255
115 permit 10.0.0.0, wildcard bits 0.0.0.255
130 permit 10.5.5.0, wildcard bits 0.0.0.255
145 permit 10.0.0.0, wildcard bits 0.0.0.255
```

Configuring Commented IP ACL Entries

Either use a named or numbered access list configuration. You must apply the access list to an interface or terminal line after the access list is created for the configuration to work.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip access-list** {standard | extended} {name | number}
4. **remark** *remark*
5. **deny protocol host** *host-address* **any eq port**
6. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.

	Command or Action	Purpose
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	ip access-list {standard extended} {name number} Example: Device(config)# ip access-list extended telnetting	Identifies the access list by a name or number and enters extended named access list configuration mode.
Step 4	remark remark Example: Device(config-ext-nacl)# remark Do not allow host1 subnet to telnet out	Adds a remark for an entry in a named IP access list. <ul style="list-style-type: none"> The remark indicates the purpose of the permit or deny statement.
Step 5	deny protocol host host-address any eq port Example: Device(config-ext-nacl)# deny tcp host 172.16.2.88 any eq telnet	Sets conditions in a named IP access list that denies packets.
Step 6	end Example: Device(config-ext-nacl)# end	Exits extended named access list configuration mode and enters privileged EXEC mode.

Configuring Time Ranges for ACLs

Follow these steps to configure a time-range parameter for an ACL:

SUMMARY STEPS

- enable**
- configure terminal**
- time-range** *time-range-name*
- Use one of the following:
 - absolute** [start *time date*] [end *time date*]
 - periodic** *day-of-the-week hh:mm to [day-of-the-week] hh:mm*
 - periodic** {weekdays | weekend | daily} *hh:mm to hh:mm*
- end**
- show running-config**
- copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device(config)# enable	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	time-range <i>time-range-name</i> Example: Device(config)# time-range workhours	Assigns a meaningful name (for example, <i>workhours</i>) to the time range to be created, and enter time-range configuration mode. The name cannot contain a space or quotation mark and must begin with a letter.
Step 4	Use one of the following: <ul style="list-style-type: none"> • absolute [start <i>time date</i>] [end <i>time date</i>] • periodic <i>day-of-the-week hh:mm to [day-of-the-week] hh:mm</i> • periodic {weekdays weekend daily} <i>hh:mm to hh:mm</i> Example: Device(config-time-range)# absolute start 00:00 1 Jan 2006 end 23:59 1 Jan 2006 OR Device(config-time-range)# periodic weekdays 8:00 to 12:00	Specifies when the function it will be applied to is operational. <ul style="list-style-type: none"> • You can use only one absolute statement in the time range. If you configure more than one absolute statement, only the one configured last is executed. • You can enter multiple periodic statements. For example, you could configure different hours for weekdays and weekends. See the example configurations.
Step 5	end Example: Device(config)# end	Returns to privileged EXEC mode.
Step 6	show running-config Example: Device# show running-config	Verifies your entries.

	Command or Action	Purpose
Step 7	copy running-config startup-config Example: Device# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

What to do next

Repeat the steps if you have multiple items that you want in effect at different times.

Applying an IPv4 ACL to a Terminal Line

You can use numbered ACLs to control access to one or more terminal lines. You cannot apply named ACLs to lines. You must set identical restrictions on all the virtual terminal lines because a user can attempt to connect to any of them.

Follow these steps to restrict incoming and outgoing connections between a virtual terminal line and the addresses in an ACL:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **line [console | vty] *line-number***
4. **access-class *access-list-number* {in | out}**
5. **end**
6. **show running-config**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device(config)# <code>enable</code>	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode.
Step 3	line [console vty] <i>line-number</i> Example:	Identifies a specific line to configure, and enter in-line configuration mode.

	Command or Action	Purpose
	Device(config)# line console 0	<ul style="list-style-type: none"> • console—Specifies the console terminal line. The console port is DCE. • vty—Specifies a virtual terminal for remote console access. <p>The <i>line-number</i> is the first line number in a contiguous group that you want to configure when the line type is specified. The range is from 0 to 16.</p>
Step 4	access-class <i>access-list-number</i> { in out } Example: Device(config-line)# access-class 10 in	Restricts incoming and outgoing connections between a particular virtual terminal line (into a device) and the addresses in an access list.
Step 5	end Example: Device(config-line)# end	Returns to privileged EXEC mode.
Step 6	show running-config Example: Device# show running-config	Verifies your entries.
Step 7	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Applying an IPv4 ACL to an Interface (CLI)

This section describes how to apply IPv4 ACLs to network interfaces.

Beginning in privileged EXEC mode, follow the procedure given below to control access to an interface:

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **ip access-group** {*access-list-number* | *name*} {**in** | **out**}
4. **end**
5. **show running-config**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode.
Step 2	interface <i>interface-id</i> Example: Device(config)# <code>interface gigabitethernet1/0/1</code>	Identifies a specific interface for configuration, and enter interface configuration mode. The interface can be a Layer 2 interface (port ACL), or a Layer 3 interface (router ACL).
Step 3	ip access-group {<i>access-list-number</i> <i>name</i>} {in out} Example: Device(config-if)# <code>ip access-group 2 in</code>	Controls access to the specified interface.
Step 4	end Example: Device(config-if)# <code>end</code>	Returns to privileged EXEC mode.
Step 5	show running-config Example: Device# <code>show running-config</code>	Displays the access list configuration.
Step 6	copy running-config startup-config Example: Device# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Monitoring IPv4 ACLs

You can monitor IPv4 ACLs by displaying the ACLs that are configured on the switch, and displaying the ACLs that have been applied to interfaces and VLANs.

When you use the **ip access-group** interface configuration command to apply ACLs to a Layer 2 or 3 interface, you can display the access groups on the interface. You can also display the MAC ACLs applied to a Layer 2 interface. You can use the privileged EXEC commands as described in this table to display this information.

Table 134: Commands for Displaying Access Lists and Access Groups

Command	Purpose
show access-lists [<i>number</i> <i>name</i>]	Displays the contents of one or all current IP and MAC address access lists or a specific access list (numbered or named).
show ip access-lists [<i>number</i> <i>name</i>]	Displays the contents of all current IP access lists or a specific IP access list (numbered or named).
show ip interface <i>interface-id</i>	Displays detailed configuration and status of an interface. If IP is enabled on the interface and ACLs have been applied by using the ip access-group interface configuration command, the access groups are included in the display.
show running-config [interface <i>interface-id</i>]	Displays the contents of the configuration file for the switch or the specified interface, including all configured MAC and IP access lists and which access groups are applied to an interface.
show mac access-group [interface <i>interface-id</i>]	Displays MAC access lists applied to all Layer 2 interfaces or the specified Layer 2 interface.

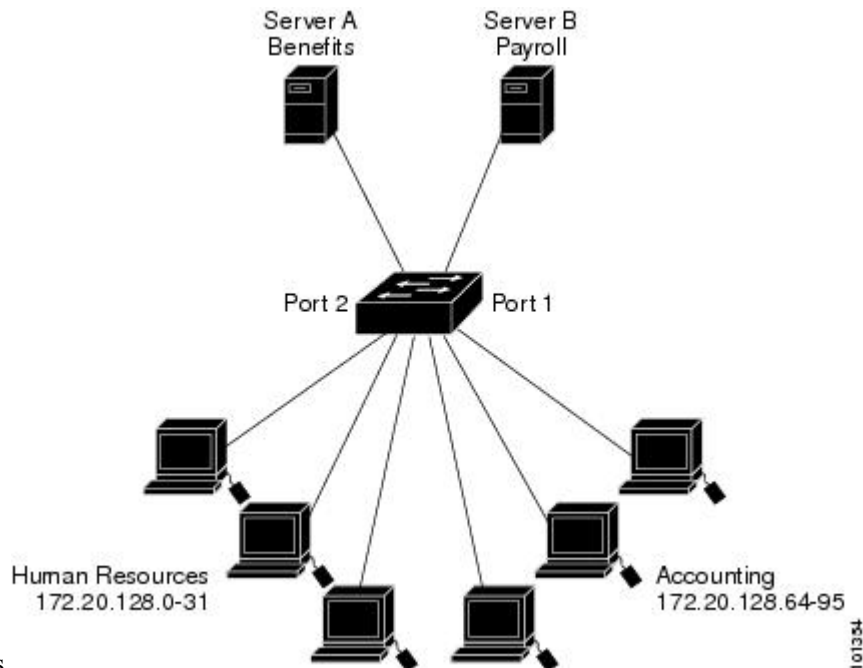
Configuration Examples for ACLs

ACLs in a Small Networked Office

Figure 96: Using Router ACLs to Control Traffic

This shows a small networked office environment with routed Port 2 connected to Server A, containing benefits and other information that all employees can access, and routed Port 1 connected to Server B, containing

confidential payroll data. All users can access Server A, but Server B has restricted



access.

Use router ACLs to do this in one of two ways:

- Create a standard ACL, and filter traffic coming to the server from Port 1.
- Create an extended ACL, and filter traffic coming from the server into Port 1.

Example: Numbered ACLs

In this example, network 10.0.0.0 is a Class A network whose second octet specifies a subnet; that is, its subnet mask is 255.255.0.0. The third and fourth octets of a network 10.0.0.0 address specify a particular host. Using access list 2, the switch accepts one address on subnet 48 and reject all others on that subnet. The last line of the list shows that the switch accepts addresses on all other network 10.0.0.0 subnets. The ACL is applied to packets entering a port.

```
Device(config)# access-list 2 permit 10.48.0.3
Device(config)# access-list 2 deny 10.48.0.0 0.0.255.255
Device(config)# access-list 2 permit 10.0.0.0 0.255.255.255
Device(config)# interface gigabitEthernet2/0/1
Device(config-if)# ip access-group 2 in
```

Examples: Extended ACLs

In this example, the first line permits any incoming TCP connections with destination ports greater than 1023. The second line permits incoming TCP connections to the Simple Mail Transfer Protocol (SMTP) port of host 128.88.1.2. The third line permits incoming ICMP messages for error feedback.

```
Device(config)# access-list 102 permit tcp any 128.88.0.0 0.0.255.255 gt 1023
Device(config)# access-list 102 permit tcp any host 128.88.1.2 eq 25
```

```
Device(config)# access-list 102 permit icmp any any
Device(config)# interface gigabitethernet2/0/1
Device(config-if)# ip access-group 102 in
```

In this example, suppose that you have a network connected to the Internet, and you want any host on the network to be able to form TCP connections to any host on the Internet. However, you do not want IP hosts to be able to form TCP connections to hosts on your network, except to the mail (SMTP) port of a dedicated mail host.

SMTP uses TCP port 25 on one end of the connection and a random port number on the other end. The same port numbers are used throughout the life of the connection. Mail packets coming in from the Internet have a destination port of 25. Outbound packets have the port numbers reversed. Because the secure system of the network always accepts mail connections on port 25, the incoming and outgoing services are separately controlled. The ACL must be configured as an input ACL on the outbound interface and an output ACL on the inbound interface.

```
Device(config)# access-list 102 permit tcp any 128.88.0.0 0.0.255.255 eq 23
Device(config)# access-list 102 permit tcp any 128.88.0.0 0.0.255.255 eq 25
Device(config)# interface gigabitethernet1/0/1
Device(config-if)# ip access-group 102 in
```

In this example, the network is a Class B network with the address 128.88.0.0, and the mail host address is 128.88.1.2. The **established** keyword is used only for the TCP to show an established connection. A match occurs if the TCP datagram has the ACK or RST bits set, which show that the packet belongs to an existing connection. Gigabit Ethernet interface 1 on stack member 1 is the interface that connects the router to the Internet.

```
Device(config)# access-list 102 permit tcp any 128.88.0.0 0.0.255.255 established
Device(config)# access-list 102 permit tcp any host 128.88.1.2 eq 25
Device(config)# interface gigabitethernet1/0/1
Device(config-if)# ip access-group 102 in
```

Examples: Named ACLs

Creating named standard and extended ACLs

This example creates a standard ACL named *internet_filter* and an extended ACL named *marketing_group*. The *internet_filter* ACL allows all traffic from the source address 1.2.3.4.

```
Device(config)# ip access-list standard Internet_filter
Device(config-ext-nacl)# permit 1.2.3.4
Device(config-ext-nacl)# exit
```

The *marketing_group* ACL allows any TCP Telnet traffic to the destination address and wildcard 171.69.0.0 0.0.255.255 and denies any other TCP traffic. It permits ICMP traffic, denies UDP traffic from any source to the destination address range 171.69.0.0 through 179.69.255.255 with a destination port less than 1024, denies any other IP traffic, and provides a log of the result.

```
Device(config)# ip access-list extended marketing_group
Device(config-ext-nacl)# permit tcp any 171.69.0.0 0.0.255.255 eq telnet
Device(config-ext-nacl)# deny tcp any any
```

Example: Configuring an Access Control Entry with Noncontiguous Ports

```
Device(config-ext-nacl)# permit icmp any any
Device(config-ext-nacl)# deny udp any 171.69.0.0 0.0.255.255 lt 1024
Device(config-ext-nacl)# deny ip any any log
Device(config-ext-nacl)# exit
```

The *Internet_filter* ACL is applied to outgoing traffic and the *marketing_group* ACL is applied to incoming traffic on a Layer 3 port.

```
Device(config)# interface gigabitethernet3/0/1
Device(config-if)# no switchport
Device(config-if)# ip address 2.0.5.1 255.255.255.0
Device(config-if)# ip access-group Internet_filter out
Device(config-if)# ip access-group marketing_group in
```

Deleting individual ACEs from named ACLs

This example shows how you can delete individual ACEs from the named access list *border-list*:

```
Device(config)# ip access-list extended border-list
Device(config-ext-nacl)# no permit ip host 10.1.1.3 any
```

Example: Configuring an Access Control Entry with Noncontiguous Ports

The following access list entry can be created because up to ten ports can be entered after the **eq** and **neq** operators:

```
ip access-list extended aaa
 permit tcp any eq telnet ftp any eq 23 45 34
end
```

Enter the **show access-lists** command to display the newly created access list entry.

```
Device# show access-lists aaa

Extended IP access list aaa
 10 permit tcp any eq telnet ftp any eq 23 45 34
```

Example: Consolidating Access List Entries with Noncontiguous Ports into One Access List Entry

The **show access-lists** command is used to display a group of access list entries for the access list named abc:

```
Device# show access-lists abc
Extended IP access list abc
 10 permit tcp any eq telnet any eq 450
 20 permit tcp any eq telnet any eq 679
 30 permit tcp any eq ftp any eq 450
 40 permit tcp any eq ftp any eq 679
```

Because the entries are all for the same **permit** statement and simply show different ports, they can be consolidated into one new access list entry. The following example shows the removal of the redundant access list entries and the creation of a new access list entry that consolidates the previously displayed group of access list entries:

```
ip access-list extended abc
```

```
no 10
no 20
no 30
no 40
permit tcp any eq telnet ftp any eq 450 679
end
```

When the **show access-lists** command is reentered, the consolidated access list entry is displayed:

```
Device# show access-lists abc
Extended IP access list abc
 10 permit tcp any eq telnet ftp any eq 450 679
```

Example Resequencing Entries in an Access List

The following example shows an access list before and after resequencing. The starting value is 1, and increment value is 2. The subsequent entries are ordered based on the increment values that users provide, and the range is from 1 to 2147483647.

When an entry with no sequence number is entered, by default it has a sequence number of 10 more than the last entry in the access list.

```
Router# show access-list carls
Extended IP access list carls
 10 permit ip host 10.3.3.3 host 172.16.5.34
 20 permit icmp any any
 30 permit tcp any host 10.3.3.3
 40 permit ip host 10.4.4.4 any
 50 Dynamic test permit ip any any
 60 permit ip host 172.16.2.2 host 10.3.3.12
 70 permit ip host 10.3.3.3 any log
 80 permit tcp host 10.3.3.3 host 10.1.2.2
 90 permit ip host 10.3.3.3 any
100 permit ip any any
Router(config)# ip access-list extended carls
Router(config)# ip access-list resequence carls 1 2
Router(config)# end
Router# show access-list carls
Extended IP access list carls
 1 permit ip host 10.3.3.3 host 172.16.5.34
 3 permit icmp any any
 5 permit tcp any host 10.3.3.3
 7 permit ip host 10.4.4.4 any
 9 Dynamic test permit ip any any
11 permit ip host 172.16.2.2 host 10.3.3.12
13 permit ip host 10.3.3.3 any log
15 permit tcp host 10.3.3.3 host 10.1.2.2
17 permit ip host 10.3.3.3 any
19 permit ip any any
```

Example Adding an Entry with a Sequence Number

In the following example, a new entry (sequence number 15) is added to an access list:

```
Router# show ip access-list
Standard IP access list tryon
 2 permit 10.4.4.2, wildcard bits 0.0.255.255
 5 permit 10.0.0.44, wildcard bits 0.0.0.255
```

```

10 permit 10.0.0.1, wildcard bits 0.0.0.255
20 permit 10.0.0.2, wildcard bits 0.0.0.255
Router(config)# ip access-list standard tryon
Router(config-std-nacl)# 15 permit 10.5.5.5 0.0.0.255
Router# show ip access-list
Standard IP access list tryon
2 permit 10.4.0.0, wildcard bits 0.0.255.255
5 permit 10.0.0.0, wildcard bits 0.0.0.255
10 permit 10.0.0.0, wildcard bits 0.0.0.255
15 permit 10.5.5.0, wildcard bits 0.0.0.255
20 permit 10.0.0.0, wildcard bits 0.0.0.255

```

Example Adding an Entry with No Sequence Number

The following example shows how an entry with no specified sequence number is added to the end of an access list. When an entry is added without a sequence number, it is automatically given a sequence number that puts it at the end of the access list. Because the default increment is 10, the entry will have a sequence number 10 higher than the last entry in the existing access list.

```

Router(config)# ip access-list standard resources
Router(config-std-nacl)# permit 10.1.1.1 0.0.0.255
Router(config-std-nacl)# permit 10.2.2.2 0.0.0.255
Router(config-std-nacl)# permit 10.3.3.3 0.0.0.255
Router# show access-list
Standard IP access list resources
10 permit 10.1.1.1, wildcard bits 0.0.0.255
20 permit 10.2.2.2, wildcard bits 0.0.0.255
30 permit 10.3.3.3, wildcard bits 0.0.0.255
Router(config)# ip access-list standard resources
Router(config-std-nacl)# permit 10.4.4.4 0.0.0.255
Router(config-std-nacl)# end
Router# show access-list
Standard IP access list resources
10 permit 10.1.1.1, wildcard bits 0.0.0.255
20 permit 10.2.2.2, wildcard bits 0.0.0.255
30 permit 10.3.3.3, wildcard bits 0.0.0.255
40 permit 10.4.4.4, wildcard bits 0.0.0.255

```

Examples: Configuring Commented IP ACL Entries

In this example of a numbered ACL, the workstation that belongs to Jones is allowed access, and the workstation that belongs to Smith is not allowed access:

```

Device(config)# access-list 1 remark Permit only Jones workstation through
Device(config)# access-list 1 permit 171.69.2.88
Device(config)# access-list 1 remark Do not allow Smith workstation through
Device(config)# access-list 1 deny 171.69.3.13

```

In this example of a numbered ACL, the Winter and Smith workstations are not allowed to browse the web:

```

Device(config)# access-list 100 remark Do not allow Winter to browse the web
Device(config)# access-list 100 deny host 171.69.3.85 any eq www
Device(config)# access-list 100 remark Do not allow Smith to browse the web
Device(config)# access-list 100 deny host 171.69.3.13 any eq www

```


In this example of a named ACL, the Jones subnet is not allowed access:

```
Device(config)# ip access-list standard prevention
Device(config-std-nacl)# remark Do not allow Jones subnet through
Device(config-std-nacl)# deny 171.69.0.0 0.0.255.255
```

In this example of a named ACL, the Jones subnet is not allowed to use outbound Telnet:

```
Device(config)# ip access-list extended telnetting
Device(config-ext-nacl)# remark Do not allow Jones subnet to telnet out
Device(config-ext-nacl)# deny tcp 171.69.0.0 0.0.255.255 any eq telnet
```

Examples: Using Time Ranges with ACLs

This example shows how to verify after you configure time ranges for *workhours* and to configure January 1, 2006, as a company holiday.

```
Device# show time-range
time-range entry: new_year_day_2003 (inactive)
    absolute start 00:00 01 January 2006 end 23:59 01 January 2006
time-range entry: workhours (inactive)
    periodic weekdays 8:00 to 12:00
    periodic weekdays 13:00 to 17:00
```

To apply a time range, enter the time-range name in an extended ACL that can implement time ranges. This example shows how to create and verify extended access list 188 that denies TCP traffic from any source to any destination during the defined holiday times and permits all TCP traffic during work hours.

```
Device(config)# access-list 188 deny tcp any any time-range new_year_day_2006
Device(config)# access-list 188 permit tcp any any time-range workhours
Device(config)# end
Device# show access-lists
Extended IP access list 188
    10 deny tcp any any time-range new_year_day_2006 (inactive)
    20 permit tcp any any time-range workhours (inactive)
```

This example uses named ACLs to permit and deny the same traffic.

```
Device(config)# ip access-list extended deny_access
Device(config-ext-nacl)# deny tcp any any time-range new_year_day_2006
Device(config-ext-nacl)# exit
Device(config)# ip access-list extended may_access
Device(config-ext-nacl)# permit tcp any any time-range workhours
Device(config-ext-nacl)# end
Device# show ip access-lists
Extended IP access list lpip_default
    10 permit ip any any
Extended IP access list deny_access
    10 deny tcp any any time-range new_year_day_2006 (inactive)
Extended IP access list may_access
    10 permit tcp any any time-range workhours (inactive)
```

Examples: Time Range Applied to an IP ACL

This example denies HTTP traffic on IP on Monday through Friday between the hours of 8:00 a.m. and 6:00 p.m (18:00). The example allows UDP traffic only on Saturday and Sunday from noon to 8:00 p.m. (20:00).

```
Device(config)# time-range no-http
Device(config)# periodic weekdays 8:00 to 18:00
!
Device(config)# time-range udp-yes
Device(config)# periodic weekend 12:00 to 20:00
!
Device(config)# ip access-list extended strict
Device(config-ext-nacl)# deny tcp any any eq www time-range no-http
Device(config-ext-nacl)# permit udp any any time-range udp-yes
!
Device(config-ext-nacl)# exit
Device(config)# interface gigabitethernet2/0/1
Device(config-if)# ip access-group strict in
```

Examples: ACL Logging

Two variations of logging are supported on ACLs. The **log** keyword sends an informational logging message to the console about the packet that matches the entry; the **log-input** keyword includes the input interface in the log entry.

In this example, standard named access list *stan1* denies traffic from 10.1.1.0 0.0.0.255, allows traffic from all other sources, and includes the **log** keyword.

```
Device(config)# ip access-list standard stan1
Device(config-std-nacl)# deny 10.1.1.0 0.0.0.255 log
Device(config-std-nacl)# permit any log
Device(config-std-nacl)# exit
Device(config)# interface gigabitethernet1/0/1
Device(config-if)# ip access-group stan1 in
Device(config-if)# end
Device# show logging
Syslog logging: enabled (0 messages dropped, 0 flushes, 0 overruns)
  Console logging: level debugging, 37 messages logged
  Monitor logging: level debugging, 0 messages logged
  Buffer logging: level debugging, 37 messages logged
  File logging: disabled
  Trap logging: level debugging, 39 message lines logged

Log Buffer (4096 bytes):

00:00:48: NTP: authentication delay calculation problems

<output truncated>

00:09:34:%SEC-6-IPACCESSLOGS:list stan1 permitted 0.0.0.0 1 packet
00:09:59:%SEC-6-IPACCESSLOGS:list stan1 denied 10.1.1.15 1 packet
00:10:11:%SEC-6-IPACCESSLOGS:list stan1 permitted 0.0.0.0 1 packet
```

This example is a named extended access list *ext1* that permits ICMP packets from any source to 10.1.1.0 0.0.0.255 and denies all UDP packets.

```

Device(config)# ip access-list extended ext1
Device(config-ext-nacl)# permit icmp any 10.1.1.0 0.0.0.255 log
Device(config-ext-nacl)# deny udp any any log
Device(config-std-nacl)# exit
Device(config)# interface gigabitethernet1/0/2
Device(config-if)# ip access-group ext1 in

```

This is an example of a log for an extended ACL:

```

01:24:23:%SEC-6-IPACCESSLOGDP:list ext1 permitted icmp 10.1.1.15 -> 10.1.1.61 (0/0), 1
packet
01:25:14:%SEC-6-IPACCESSLOGDP:list ext1 permitted icmp 10.1.1.15 -> 10.1.1.61 (0/0), 7
packets
01:26:12:%SEC-6-IPACCESSLOGP:list ext1 denied udp 0.0.0.0(0) -> 255.255.255.255(0), 1 packet
01:31:33:%SEC-6-IPACCESSLOGP:list ext1 denied udp 0.0.0.0(0) -> 255.255.255.255(0), 8 packets

```

Note that all logging entries for IP ACLs start with %SEC-6-IPACCESSLOG with minor variations in format depending on the kind of ACL and the access entry that has been matched.

This is an example of an output message when the **log-input** keyword is entered:

```

00:04:21:%SEC-6-IPACCESSLOGDP:list inputlog permitted icmp 10.1.1.10 (Vlan1 0001.42ef.a400)
->
10.1.1.61 (0/0), 1 packet

```

A log message for the same sort of packet using the **log** keyword does not include the input interface information:

```

00:05:47:%SEC-6-IPACCESSLOGDP:list inputlog permitted icmp 10.1.1.10 -> 10.1.1.61 (0/0), 1
packet

```

Examples: Troubleshooting ACLs

If this ACL manager message appears and [chars] is the access-list name,

```

ACLMGR-2-NOVMR: Cannot generate hardware representation of access list [chars]

```

The switch has insufficient resources to create a hardware representation of the ACL. The resources include hardware memory and label space but not CPU memory. A lack of available logical operation units or specialized hardware resources causes this problem. Logical operation units are needed for a TCP flag match or a test other than **eq** (**ne**, **gt**, **lt**, or **range**) on TCP, UDP, or SCTP port numbers.

Use one of these workarounds:

- Modify the ACL configuration to use fewer resources.
- Rename the ACL with a name or number that alphanumerically precedes the ACL names or numbers.

To determine the specialized hardware resources, enter the **show platform layer4 acl** map privileged EXEC command. If the switch does not have available resources, the output shows that index 0 to index 15 are not available.

For more information about configuring ACLs with insufficient resources, see CSCsq63926 in the Bug Toolkit.

For example, if you apply this ACL to an interface:

```
permit tcp source source-wildcard destination destination-wildcard range 5 60
permit tcp source source-wildcard destination destination-wildcard range 15 160
permit tcp source source-wildcard destination destination-wildcard range 115 1660
permit tcp source source-wildcard destination destination-wildcard
```

And if this message appears:

```
ACL MGR-2-NOVMR: Cannot generate hardware representation of access list [chars]
```

The flag-related operators are not available. To avoid this issue,

- Move the fourth ACE before the first ACE by using **ip access-list resequence** global configuration command:

```
permit tcp source source-wildcard destination destination-wildcard
permit tcp source source-wildcard destination destination-wildcard range 5 60
permit tcp source source-wildcard destination destination-wildcard range 15 160
permit tcp source source-wildcard destination destination-wildcard range 115 1660
```

or

- Rename the ACL with a name or number that alphanumerically precedes the other ACLs (for example, rename ACL 79 to ACL 1).

You can now apply the first ACE in the ACL to the interface. The switch allocates the ACE to available mapping bits in the Opselect index and then allocates flag-related operators to use the same bits in the hardware memory.

Additional References

Related Documents

MIBs

MIB	MIBs Link
All the supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
<p>The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies.</p> <p>To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds.</p> <p>Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.</p>	<p>http://www.cisco.com/support</p>

Feature Information for IPv4 Access Control Lists

Release	Feature Information
Cisco IOS Release 15.0(2)EX1	IPv4 Access Control Lists perform packet filtering to control which packets move through the network and where. Such control provides security by helping to limit network traffic, restrict the access of users and devices to the network, and prevent traffic from leaving a network. This feature was introduced.
Cisco IOS 15.2(2)E	The Named ACL Support for Noncontiguous Ports on an Access Control Entry feature allows you to specify noncontiguous ports in a single access control entry, which greatly reduces the number of entries required in an access control list when several entries have the same source address, destination address, and protocol, but differ only in the ports.
Cisco IOS 15.2(2)E	<p>The IP Access List Entry Sequence Numbering feature helps users to apply sequence numbers to permit or deny statements and also reorder, add, or remove such statements from a named IP access list. This feature makes revising IP access lists much easier. Prior to this feature, users could add access list entries to the end of an access list only; therefore needing to add statements anywhere except the end required reconfiguring the access list entirely.</p> <p>The following commands were introduced or modified: deny (IP), ip access-list resequence deny (IP), permit (IP).</p>



CHAPTER 68

IPv6 Access Control Lists

Access lists determine what traffic is blocked and what traffic is forwarded at device interfaces and allow filtering of traffic based on source and destination addresses, and inbound and outbound traffic to a specific interface. Standard IPv6 ACL functionality was extended to support traffic filtering based on IPv6 option headers and optional, upper-layer protocol type information for finer granularity of control. Standard IPv6 ACL functionality was extended to support traffic filtering based on IPv6 option headers and optional, upper-layer protocol type information for finer granularity of control.

This module describes how to configure IPv6 traffic filtering and to control access to virtual terminal lines.

- [Prerequisites for IPv6 ACLs, on page 1349](#)
- [Restrictions for IPv6 ACLs, on page 1349](#)
- [Information About Configuring IPv6 ACLs, on page 1350](#)
- [How to Configure IPv6 ACLs, on page 1353](#)
- [Configuration Examples for IPv6 ACLs, on page 1362](#)
- [Additional References, on page 1363](#)
- [Feature Information for IPv6 Access Control Lists, on page 1364](#)

Prerequisites for IPv6 ACLs

The following are the prerequisites for IPv6 ACLs:

To use IPv6, you must configure the dual IPv4 and IPv6 Switch Database Management (SDM) template on the switch. You select the template by entering the **sdm prefer {default }** global configuration command.

Restrictions for IPv6 ACLs

With IPv4, you can configure standard and extended numbered IP ACLs, named IP ACLs, and MAC ACLs. IPv6 supports only named ACLs.

The switch supports most Cisco IOS-supported IPv6 ACLs with some exceptions:

- The switch does not support matching on these keywords: **routing header**, and **undetermined-transport**.
- The switch does not support reflexive ACLs (the **reflect** keyword).
-
-

- This release supports port ACLs, router ACLs and VLAN ACLs (VLAN maps) for IPv6.
-
- The switch does not apply MAC-based ACLs on IPv6 frames.
- When configuring an ACL, there is no restriction on keywords entered in the ACL, regardless of whether or not they are supported on the platform. When you apply the ACL to an interface that requires hardware forwarding (physical ports or SVIs), the switch checks to determine whether or not the ACL can be supported on the interface. If not, attaching the ACL is rejected.
- If an ACL is applied to an interface and you attempt to add an access control entry (ACE) with an unsupported keyword, the switch does not allow the ACE to be added to the ACL that is currently attached to the interface.

IPv6 ACLs on the switch have these characteristics:

- Fragmented frames (the **fragments** keyword as in IPv4) are supported
- The same statistics supported in IPv4 are supported for IPv6 ACLs.
- If the switch runs out of hardware space, the packets associated with the ACL are dropped on the interface.
- Logging is supported for router ACLs, but not for port ACLs.
- The switch supports IPv6 address-matching for a full range of prefix-lengths.

Information About Configuring IPv6 ACLs

You can filter IP version 6 (IPv6) traffic by creating IPv6 access control lists (ACLs) and applying them to interfaces similarly to the way that you create and apply IP version 4 (IPv4) named ACLs. You can also create and apply input router ACLs to filter Layer 3 management traffic.



Note To use IPv6, you must configure the dual IPv4 and IPv6 Switch Database Management (SDM) template on the switch. You select the template by entering the **sdm prefer {default }** global configuration command.

ACL Overview

Packet filtering can help limit network traffic and restrict network use by certain users or devices. ACLs filter traffic as it passes through a router or switch and permit or deny packets crossing specified interfaces or VLANs. An ACL is a sequential collection of permit and deny conditions that apply to packets. When a packet is received on an interface, the switch compares the fields in the packet against any applied ACLs to verify that the packet has the required permissions to be forwarded, based on the criteria specified in the access lists. One by one, it tests packets against the conditions in an access list. The first match decides whether the switch accepts or rejects the packets. Because the switch stops testing after the first match, the order of conditions in the list is critical. If no conditions match, the switch rejects the packet. If there are no restrictions, the switch forwards the packet; otherwise, the switch drops the packet. The switch can use ACLs on all packets it forwards, including packets bridged within a VLAN.

You configure access lists on a router or Layer 3 switch to provide basic security for your network. If you do not configure ACLs, all packets passing through the switch could be allowed onto all parts of the network.

You can use ACLs to control which hosts can access different parts of a network or to decide which types of traffic are forwarded or blocked at router interfaces. For example, you can allow e-mail traffic to be forwarded but not Telnet traffic. ACLs can be configured to block inbound traffic, outbound traffic, or both.

IPv6 ACLs Overview

You can filter IP Version 6 (IPv6) traffic by creating IPv6 access control lists (ACLs) and applying them to interfaces similar to how you create and apply IP Version 4 (IPv4) named ACLs. You can also create and apply input router ACLs to filter Layer 3 management traffic when the switch is running IP base and LAN base feature sets.

A switch supports three types of IPv6 ACLs:

- IPv6 router ACLs are supported on outbound or inbound traffic on Layer 3 interfaces, which can be routed ports, switch virtual interfaces (SVIs), or Layer 3 EtherChannels. IPv6 router ACLs apply only to IPv6 packets that are routed.
- IPv6 port ACLs are supported on outbound and inbound Layer 2 interfaces. IPv6 port ACLs are applied to all IPv6 packets entering the interface.
- VLAN ACLs or VLAN maps access-control all packets in a VLAN. You can use VLAN maps to filter traffic between devices in the same VLAN. ACL VLAN maps are applied on L2 VLANs. VLAN maps are configured to provide access control based on Layer 3 addresses for IPv6. Unsupported protocols are access-controlled through MAC addresses using Ethernet ACEs. After a VLAN map is applied to a VLAN, all packets entering the VLAN are checked against the VLAN map.

The switch supports VLAN ACLs (VLAN maps) for IPv6 traffic.

You can apply both IPv4 and IPv6 ACLs to an interface. As with IPv4 ACLs, IPv6 port ACLs take precedence over router ACLs.

Understanding IPv6 ACLs

A switch image supports two types of IPv6 ACLs:

- IPv6 router ACLs - Supported on inbound or outbound traffic on Layer 3 interfaces, which can be routed ports, switch virtual interfaces (SVIs), or Layer 3 EtherChannels. Applied to only IPv6 packets that are routed.
- IPv6 port ACLs - Supported on inbound traffic on Layer 2 interfaces only. Applied to all IPv6 packets entering the interface.



Note If you configure unsupported IPv6 ACLs, an error message appears and the configuration does not take effect.

The switch does not support VLAN ACLs (VLAN maps) for IPv6 traffic.

You can apply both IPv4 and IPv6 ACLs to an interface.

As with IPv4 ACLs, IPv6 port ACLs take precedence over router ACLs:

- When an input router ACL and input port ACL exist in an SVI, packets received on ports to which a port ACL is applied are filtered by the port ACL. Routed IP packets received on other ports are filtered by the router ACL. Other packets are not filtered.

- When an output router ACL and input port ACL exist in an SVI, packets received on the ports to which a port ACL is applied are filtered by the port ACL. Outgoing routed IPv6 packets are filtered by the router ACL. Other packets are not filtered.



Note If any port ACL (IPv4, IPv6, or MAC) is applied to an interface, that port ACL is used to filter packets, and any router ACLs attached to the SVI of the port VLAN are ignored.

Interactions with Other Features and Switches

- If an IPv6 router ACL is configured to deny a packet, the packet is not routed. A copy of the packet is sent to the Internet Control Message Protocol (ICMP) queue to generate an ICMP unreachable message for the frame.
- If a bridged frame is to be dropped due to a port ACL, the frame is not bridged.
- You can create both IPv4 and IPv6 ACLs on a switch or switch stack, and you can apply both IPv4 and IPv6 ACLs to the same interface. Each ACL must have a unique name; an error message appears if you try to use a name that is already configured.

You use different commands to create IPv4 and IPv6 ACLs and to attach IPv4 or IPv6 ACLs to the same Layer 2 or Layer 3 interface. If you use the wrong command to attach an ACL (for example, an IPv4 command to attach an IPv6 ACL), you receive an error message.

- You cannot use MAC ACLs to filter IPv6 frames. MAC ACLs can only filter non-IP frames.
- If the hardware memory is full, packets are dropped on the interface and an unload error message is logged.

Default Configuration for IPv6 ACLs

The default IPv6 ACL configuration is as follows:

```
Switch# show access-lists preauth_ipv6_acl
IPv6 access list preauth_ipv6_acl (per-user)
permit udp any any eq domain sequence 10
permit tcp any any eq domain sequence 20
permit icmp any any nd-ns sequence 30
permit icmp any any nd-na sequence 40
permit icmp any any router-solicitation sequence 50
permit icmp any any router-advertisement sequence 60
permit icmp any any redirect sequence 70
permit udp any eq 547 any eq 546 sequence 80
permit udp any eq 546 any eq 547 sequence 90
deny ipv6 any any sequence 100
```

Supported ACL Features

IPv6 ACLs on the switch have these characteristics:

- Fragmented frames (the fragments keyword as in IPv4) are supported.
- The same statistics supported in IPv4 are supported for IPv6 ACLs.

- If the switch runs out of TCAM space, packets associated with the ACL label are forwarded to the CPU, and the ACLs are applied in software.
- Routed or bridged packets with hop-by-hop options have IPv6 ACLs applied in software.
- Logging is supported for router ACLs, but not for port ACLs.

IPv6 Port-Based Access Control List Support

The IPv6 PACL feature provides the ability to provide access control (permit or deny) on Layer 2 switch ports for IPv6 traffic. IPv6 PACLs are similar to IPv4 PACLs, which provide access control on Layer 2 switch ports for IPv4 traffic. They are supported only in the ingress direction and in hardware.

A PACL can filter ingress traffic on Layer 2 interfaces based on Layer 3 and Layer 4 header information or non-IP Layer 2 information.

ACLs and Traffic Forwarding

The IPv6 ACL Extensions for Hop by Hop Filtering feature allows you to control IPv6 traffic that might contain hop-by-hop extension headers. You can configure an access control list (ACL) to deny all hop-by-hop traffic or to selectively permit traffic based on protocol.

IPv6 access control lists (ACLs) determine what traffic is blocked and what traffic is forwarded at device interfaces. ACLs allow filtering based on source and destination addresses, inbound and outbound to a specific interface. Use the **ipv6 access-list** command to define an IPv6 ACL, and the **deny** and **permit** commands to configure its conditions.

The IPv6 ACL Extensions for Hop by Hop Filtering feature implements RFC 2460 to support traffic filtering in any upper-layer protocol type.

How to Configure IPv6 ACLs

Configuring IPv6 ACLs

To filter IPv6 traffic, you perform these steps:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ipv6 access-list list-name**
4. **{deny | permit} protocol {source-ipv6-prefix/prefix-length|any} host source-ipv6-address} [operator [port-number]] { destination-ipv6-prefix/prefix-length | any | host destination-ipv6-address} [operator [port-number]][dscp value] [fragments] [log] [log-input] [routing] [sequence value] [time-range name]**
5. **{deny | permit} tcp {source-ipv6-prefix/prefix-length | any | host source-ipv6-address} [operator [port-number]] {destination-ipv6-prefix/prefix-length | any | host destination-ipv6-address} [operator [port-number]] [ack] [dscp value] [established] [fin] [log] [log-input] [neg {port | protocol}] [psh] [range {port | protocol}] [rst] [routing] [sequence value] [syn] [time-range name] [urg]**

6. `{deny | permit} udp {source-ipv6-prefix/prefix-length | any | host source-ipv6-address} [operator [port-number]] {destination-ipv6-prefix/prefix-length | any | host destination-ipv6-address} [operator [port-number]] [dscp value] [log] [log-input] [neg {port | protocol}] [range {port | protocol}] [routing] [sequence value] [time-range name]`
7. `{deny | permit} icmp {source-ipv6-prefix/prefix-length | any | host source-ipv6-address} [operator [port-number]] {destination-ipv6-prefix/prefix-length | any | host destination-ipv6-address} [operator [port-number]] [icmp-type [icmp-code] | icmp-message] [dscp value] [log] [log-input] [routing] [sequence value] [time-range name]`
8. `end`
9. `show ipv6 access-list`
10. `show running-config`
11. `copy running-config startup-config`

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	{ipv6 access-list list-name} Example: Device(config)# ipv6 access-list example_acl_list	Defines an IPv6 ACL name, and enters IPv6 access list configuration mode.
Step 4	{deny permit} protocol {source-ipv6-prefix/prefix-length any host source-ipv6-address} [operator [port-number]] { destination-ipv6-prefix/ prefix-length any host destination-ipv6-address} [operator [port-number]][dscp value] [fragments] [log] [log-input] [routing] [sequence value] [time-range name]	Enter deny or permit to specify whether to deny or permit the packet if conditions are matched. These are the conditions: <ul style="list-style-type: none"> • For protocol, enter the name or number of an Internet protocol: ahp, esp, icmp, ipv6, pcp, stcp, tcp, or udp, or an integer in the range 0 to 255 representing an IPv6 protocol number. • The <i>source-ipv6-prefix/prefix-length</i> or <i>destination-ipv6-prefix/ prefix-length</i> is the source or destination IPv6 network or class of networks for which to set deny or permit conditions, specified in hexadecimal and using 16-bit values between colons (see RFC 2373). • Enter any as an abbreviation for the IPv6 prefix <code>::/0</code>.

	Command or Action	Purpose
		<ul style="list-style-type: none"> • For host <i>source-ipv6-address</i> or <i>destination-ipv6-address</i>, enter the source or destination IPv6 host address for which to set deny or permit conditions, specified in hexadecimal using 16-bit values between colons. • (Optional) For operator, specify an operand that compares the source or destination ports of the specified protocol. Operands are lt (less than), gt (greater than), eq (equal), neq (not equal), and range. If the operator follows the <i>source-ipv6-prefix/prefix-length</i> argument, it must match the source port. If the operator follows the <i>destination-ipv6- prefix/prefix-length</i> argument, it must match the destination port. • (Optional) The port-number is a decimal number from 0 to 65535 or the name of a TCP or UDP port. You can use TCP port names only when filtering TCP. You can use UDP port names only when filtering UDP. • (Optional) Enter dscp value to match a differentiated services code point value against the traffic class value in the Traffic Class field of each IPv6 packet header. The acceptable range is from 0 to 63. • (Optional) Enter fragments to check noninitial fragments. This keyword is visible only if the protocol is ipv6. • (Optional) Enter log to cause an logging message to be sent to the console about the packet that matches the entry. Enter log-input to include the input interface in the log entry. Logging is supported only for router ACLs. • (Optional) Enter routing to specify that IPv6 packets be routed. • (Optional) Enter sequence value to specify the sequence number for the access list statement. The acceptable range is from 1 to 4,294,967,295. • (Optional) Enter time-range name to specify the time range that applies to the deny or permit statement.
Step 5	<pre>{deny permit} tcp {source-ipv6-prefix/prefix-length any host source-ipv6-address} [operator [port-number]] {destination-ipv6- prefix/prefix-length </pre>	(Optional) Define a TCP access list and the access conditions.

	Command or Action	Purpose
	any host <i>destination-ipv6-address</i> } [operator [<i>port-number</i>]] [ack] [dscp <i>value</i>] [established] [fin] [log] [log-input] [neq { <i>port</i> protocol}] [psh] [range { <i>port</i> protocol}] [rst] [routing] [sequence <i>value</i>] [syn] [time-range <i>name</i>] [urg]	<p>Enter tcp for Transmission Control Protocol. The parameters are the same as those described in Step 3a, with these additional optional parameters:</p> <ul style="list-style-type: none"> • ack—Acknowledgment bit set. • established—An established connection. A match occurs if the TCP datagram has the ACK or RST bits set. • fin—Finished bit set; no more data from sender. • neq {<i>port</i> protocol}—Matches only packets that are not on a given port number. • psh—Push function bit set. • range {<i>port</i> protocol}—Matches only packets in the port number range. • rst—Reset bit set. • syn—Synchronize bit set. • urg—Urgent pointer bit set.
Step 6	{deny permit} udp { <i>source-ipv6-prefix/prefix-length</i> any host <i>source-ipv6-address</i> } [operator [<i>port-number</i>]] { <i>destination-ipv6-prefix/prefix-length</i> any host <i>destination-ipv6-address</i> } [operator [<i>port-number</i>]] [dscp <i>value</i>] [log] [log-input] [neq { <i>port</i> protocol}] [range { <i>port</i> protocol}] [routing] [sequence <i>value</i>] [time-range <i>name</i>]	<p>(Optional) Define a UDP access list and the access conditions.</p> <p>Enter udp for the User Datagram Protocol. The UDP parameters are the same as those described for TCP, except that the [operator [<i>port</i>]] port number or name must be a UDP port number or name, and the established parameter is not valid for UDP.</p>
Step 7	{deny permit} icmp { <i>source-ipv6-prefix/prefix-length</i> any host <i>source-ipv6-address</i> } [operator [<i>port-number</i>]] { <i>destination-ipv6-prefix/prefix-length</i> any host <i>destination-ipv6-address</i> } [operator [<i>port-number</i>]] [<i>icmp-type</i> [<i>icmp-code</i>] <i>icmp-message</i>] [dscp <i>value</i>] [log] [log-input] [routing] [sequence <i>value</i>] [time-range <i>name</i>]	<p>(Optional) Define an ICMP access list and the access conditions.</p> <p>Enter icmp for Internet Control Message Protocol. The ICMP parameters are the same as those described for most IP protocols in Step 1, with the addition of the ICMP message type and code parameters. These optional keywords have these meanings:</p> <ul style="list-style-type: none"> • <i>icmp-type</i>—Enter to filter by ICMP message type, a number from 0 to 255. • <i>icmp-code</i>—Enter to filter ICMP packets that are filtered by the ICMP message code type, a number from 0 to 255. • <i>icmp-message</i>—Enter to filter ICMP packets by the ICMP message type name or the ICMP message type and code name. To see a list of ICMP message type

	Command or Action	Purpose
		names and code names, use the ? key or see command reference for this release.
Step 8	end	Return to privileged EXEC mode.
Step 9	show ipv6 access-list	Verify the access list configuration.
Step 10	show running-config Example: Device# show running-config	Verifies your entries.
Step 11	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

What to do next

Attach the IPv6 ACL to an Interface

Attaching an IPv6 ACL to an Interface

You can apply an ACL to outbound or inbound traffic on Layer 3 interfaces, or to inbound traffic on Layer 2 interfaces. You can also apply ACLs only to inbound management traffic on Layer 3 interfaces.

Follow these steps to control access to an interface:

SUMMARY STEPS

1. enable
2. configure terminal
3. interface *interface-id*
4. no switchport
5. ipv6 address *ipv6-address*
6. ipv6 traffic-filter *access-list-name* {in | out}
7. end
8. show running-config
9. copy running-config startup-config

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example:	Enables privileged EXEC mode. • Enter your password if prompted.

	Command or Action	Purpose
	Device> enable	
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	interface <i>interface-id</i>	Identify a Layer 2 interface (for port ACLs) or Layer 3 interface (for router ACLs) on which to apply an access list, and enter interface configuration mode.
Step 4	no switchport	If applying a router ACL, this changes the interface from Layer 2 mode (the default) to Layer 3 mode.
Step 5	ipv6 address <i>ipv6-address</i>	Configure an IPv6 address on a Layer 3 interface (for router ACLs).
Step 6	ipv6 traffic-filter <i>access-list-name</i> { in out }	Apply the access list to incoming or outgoing traffic on the interface.
Step 7	end Example: Device(config)# end	Returns to privileged EXEC mode.
Step 8	show running-config Example: Device# show running-config	Verifies your entries.
Step 9	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Monitoring IPv6 ACLs

You can display information about all configured access lists, all IPv6 access lists, or a specific access list by using one or more of the privileged EXEC commands shown in the table below:

Command	Purpose
show access-lists	Displays all access lists configured on the switch.

Command	Purpose
show ipv6 access-list [<i>access-list-name</i>]	Displays all configured IPv6 access lists or the access list specified by name.
show vlan access-map [<i>map-name</i>]	Displays VLAN access map configuration.
show vlan filter [access-map <i>access-map</i> vlan <i>vlan-id</i>]	Displays the mapping between VACLs and VLANs.

This is an example of the output from the `show access-lists` privileged EXEC command. The output shows all access lists that are configured on the switch or switch stack.

```
Switch # show access-lists
Extended IP access list hello
    10 permit ip any any
IPv6 access list ipv6
    permit ipv6 any any sequence 10
```

This is an example of the output from the `show ipv6 access-list` privileged EXEC command. The output shows only IPv6 access lists configured on the switch or switch stack.

```
Switch# show ipv6 access-list
IPv6 access list inbound
    permit tcp any any eq bgp (8 matches) sequence 10
    permit tcp any any eq telnet (15 matches) sequence 20
    permit udp any any sequence 30
IPv6 access list outbound
    deny udp any any sequence 10
    deny tcp any any eq telnet sequence 20
```

This is an example of the output from the `show vlan access-map` privileged EXEC command. The output shows VLAN access map information.

```
Switch# show vlan access-map
Vlan access-map "m1" 10
  Match clauses:
    ipv6 address: ip2
  Action: drop
```

Configuring PACL Mode and Applying IPv6 PACL on an Interface

Before you begin

Before you configure the IPv6 PACL feature, you must configure an IPv6 access list. Once you have configured the IPv6 access list, you must configure the port-based access control list (PACL) mode on the specified IPv6 Layer 2 interface.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ipv6 access-list** *access-list-name*
4. **exit**

5. **interface** *type number*
6. **ipv6 traffic-filter** *access-list-name {in | out}*
7. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none">• Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	ipv6 access-list <i>access-list-name</i> Example: Device(config)# ipv6 access-list list1	Defines an IPv6 ACL and enters IPv6 access list configuration mode.
Step 4	exit Example: Device(config-ipv6-acl)# exit	Exits IPv6 access list configuration mode and enters global configuration mode.
Step 5	interface <i>type number</i> Example:	Specifies an interface type and number and enters interface configuration mode.
Step 6	ipv6 traffic-filter <i>access-list-name {in out}</i> Example: Device(config-if)# ipv6 traffic-filter list1 in	Filters incoming and outgoing IPv6 traffic on an interface.
Step 7	end Example: Device(config-if)# end	Exits interface configuration mode and enters privileged EXEC mode.

Configuring IPv6 ACL Extensions for Hop by Hop Filtering

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ipv6 access-list** *access-list-name*
4. **permit protocol** {*source-ipv6-prefix/prefix-length | any | host source-ipv6-address | auth*} [*operator [port-number]*] {*destination-ipv6-prefix/prefix-length | any | host destination-ipv6-address | auth*} [*operator [port-number]*] [**dest-option-type** [*header-number | header-type*]] [**dscp** *value*] [**flow-label** *value*] [**fragments**] [**hbh**] [**log**] [**log-input**] [**mobility**] [**mobility-type** [*mh-number | mh-type*]] [**reflect** *name*] [**timeout** *value*]] [**routing**] [**routing-type** *routing-number*] [**sequence** *value*] [**time-range** *name*]

5. **deny** *protocol* {*source-ipv6-prefix/prefix-length* | **any** | **host** *source-ipv6-address* | **auth**} [*operator* [*port-number*]] {*destination-ipv6-prefix/prefix-length* | **any** | **host** *destination-ipv6-address* | **auth**} [*operator* [*port-number*]] [**dest-option-type** [*header-number* | *header-type*]] [**dscp** *value*] [**flow-label** *value*] [**fragments**] [**hbh**] [**log**] [**log-input**] [**mobility**] [**mobility-type** [*mh-number* | *mh-type*]] [**routing**] [**routing-type** *routing-number*] [**sequence** *value*] [**time-range** *name*] [**undetermined-transport**]
6. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	ipv6 access-list <i>access-list-name</i> Example: Device(config)# ipv6 access-list hbh-acl	Defines an IPv6 ACL and enters IPv6 access list configuration mode.
Step 4	permit <i>protocol</i> { <i>source-ipv6-prefix/prefix-length</i> any host <i>source-ipv6-address</i> auth } [<i>operator</i> [<i>port-number</i>]] { <i>destination-ipv6-prefix/prefix-length</i> any host <i>destination-ipv6-address</i> auth } [<i>operator</i> [<i>port-number</i>]] [dest-option-type [<i>header-number</i> <i>header-type</i>]] [dscp <i>value</i>] [flow-label <i>value</i>] [fragments] [hbh] [log] [log-input] [mobility] [mobility-type [<i>mh-number</i> <i>mh-type</i>]] [reflect <i>name</i> [<i>timeout</i> <i>value</i>]] [routing] [routing-type <i>routing-number</i>] [sequence <i>value</i>] [time-range <i>name</i>] Example: Device(config-ipv6-acl)# permit icmp any any dest-option-type	Sets permit conditions for the IPv6 ACL.
Step 5	deny <i>protocol</i> { <i>source-ipv6-prefix/prefix-length</i> any host <i>source-ipv6-address</i> auth } [<i>operator</i> [<i>port-number</i>]] { <i>destination-ipv6-prefix/prefix-length</i> any host <i>destination-ipv6-address</i> auth } [<i>operator</i> [<i>port-number</i>]] [dest-option-type [<i>header-number</i> <i>header-type</i>]] [dscp <i>value</i>] [flow-label <i>value</i>] [fragments] [hbh] [log] [log-input] [mobility] [mobility-type [<i>mh-number</i> <i>mh-type</i>]] [routing] [routing-type <i>routing-number</i>] [sequence <i>value</i>] [time-range <i>name</i>] [undetermined-transport] Example:	Sets deny conditions for the IPv6 ACL.

	Command or Action	Purpose
	Device(config-ipv6-acl)# deny icmp any any dest-option-type	
Step 6	end Example: Device (config-ipv6-acl)# end	Returns to privileged EXEC configuration mode.

Configuration Examples for IPv6 ACLs

Example: Configuring IPv6 ACLs

This example configures the IPv6 access list named CISCO. The first deny entry in the list denies all packets that have a destination TCP port number greater than 5000. The second deny entry denies packets that have a source UDP port number less than 5000. The second deny also logs all matches to the console. The first permit entry in the list permits all ICMP packets. The second permit entry in the list permits all other traffic. The second permit entry is necessary because an implicit deny -all condition is at the end of each IPv6 access list.

```
Switch(config)# ipv6 access-list CISCO
Switch(config-ipv6-acl)# deny tcp any any gt 5000
Switch config-ipv6-acl)# deny ::/0 lt 5000 ::/0 log
Switch(config-ipv6-acl)# permit icmp any any
Switch(config-ipv6-acl)# permit any any
```

Example: Applying IPv6 ACLs

This example shows how to apply the access list Cisco to outbound traffic on a Layer 3 interface.

```
Device(config-if)# no switchport
Device(config-if)# ipv6 address 2001::/64 eui-64
Device(config-if)# ipv6 traffic-filter CISCO out
```

Example: Configuring PACL Mode and Applying IPv6 PACL on an Interface

```
Device# configure terminal
Device(config)# ipv6 access-list list1
Device(config-ipv6-acl)# exit
Device(config-if)# ipv6 traffic-filter list1 in
```

Example: IPv6 ACL Extensions for Hop by Hop Filtering

```
Device(config)# ipv6 access-list hbh_acl
Device(config-ipv6-acl)# permit tcp any any hbh
Device(config-ipv6-acl)# permit tcp any any
```

```

Device(config-ipv6-acl)# permit udp any any
Device(config-ipv6-acl)# permit udp any any hbh
Device(config-ipv6-acl)# permit hbh any any
Device(config-ipv6-acl)# permit any any
Device(config-ipv6-acl)# hardware statistics
Device(config-ipv6-acl)# exit

! Assign an IP address and add the ACL on the interface.

Device(config)# interface FastEthernet3/1
Device(config-if)# ipv6 address 1001::1/64
Device(config-if)# ipv6 traffic-filter hbh_acl in
Device(config-if)# exit
Device(config)# exit
Device# clear counters
Clear "show interface" counters on all interfaces [confirm]
Device#

! Verify the configurations.

Device# show running-config interface FastEthernet3/1

Building configuration...

Current configuration : 114 bytes
!
interface FastEthernet3/1
no switchport
ipv6 address 1001::1/64
ipv6 traffic-filter hbh_acl
end

```

Additional References

Related Documents

MIBs

MIB	MIBs Link
All the supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
<p>The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies.</p> <p>To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds.</p> <p>Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.</p>	http://www.cisco.com/support

Feature Information for IPv6 Access Control Lists

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Table 135: Feature Information for IPv6 Access Control Lists

Feature Name	Releases	Feature Information
IPv6 ACL Extensions for Hop-by-Hop Filtering	15.1(1)SG	<p>Allows you to control IPv6 traffic that might contain hop-by-hop extension headers.</p> <p>This feature was supported on CAT3560C, CAT3560CX, CAT3560X, CAT3750X, CAT4500-X.</p> <p>The following commands were introduced or modified: deny (IPv6), permit (IPv6).</p>

Feature Name	Releases	Feature Information
IPv6 PACL Support		<p>The IPv6 PACL feature permits or denies the movement of traffic between port-based interface, Layer 3 subnets, wireless or wired clients, and VLANs, or within a VLAN.</p> <p>This feature was supported on CAT2960, CAT2960S, CAT3560X, CAT3650, CAT3560CX, CAT4500.</p> <p>The following command was introduced or modified: ipv6 traffic-filter.</p>
IPv6 Services: Extended Access Control Lists	12.2(25)SG	Standard IPv6 ACL functionality was extended to support traffic filtering based on IPv6 option headers and optional, upper-layer protocol type information for finer granularity of control.
IPv6 Services: Standard Access Control Lists	12.2(25)SG	Access lists determine what traffic is blocked and what traffic is forwarded at router interfaces and allow filtering based on source and destination addresses, inbound and outbound to a specific interface.



CHAPTER 69

ACL Support for Filtering IP Options

The ACL Support for Filtering IP Options feature describes how to use an IP access list to filter IP packets that contain IP options to prevent devices from becoming saturated with spurious packets.

This module also describes the ACL TCP Flags Filtering feature and how to use an IP access list to filter IP packets that contain TCP flags. The ACL TCP Flags Filtering feature allows you to select any combination of flags on which to filter. The ability to match on a flag set and on a flag not set gives you a greater degree of control for filtering on TCP flags, thus enhancing security.

- [Prerequisites for ACL Support for Filtering IP Options, on page 1367](#)
- [Information About ACL Support for Filtering IP Options, on page 1367](#)
- [How to Configure ACL Support for Filtering IP Options, on page 1369](#)
- [Configuration Examples for ACL Support for Filtering IP Options, on page 1373](#)
- [Additional References for ACL Support for Filtering IP Options, on page 1374](#)
- [Feature Information for Creating an IP Access List to Filter, on page 1374](#)

Prerequisites for ACL Support for Filtering IP Options

Before you configure the ACL Support for Filtering IP Options feature, you must understand the concepts of the IP access lists.

Information About ACL Support for Filtering IP Options

IP Options

IP uses four key mechanisms in providing its service: Type of Service, Time to Live, Options, and Header Checksum.

The Options, commonly referred to as IP Options, provide for control functions that are required in some situations but unnecessary for the most common communications. IP Options include provisions for time stamps, security, and special routing.

IP Options may or may not appear in datagrams. They must be implemented by all IP modules (host and gateways). What is optional is their transmission in any particular datagram, not their implementation. In some environments the security option may be required in all datagrams.

The option field is variable in length. There may be zero or more options. IP Options can have one of two formats:

- Format 1: A single octet of option-type.
- Format 2: An option-type octet, an option-length octet, and the actual option-data octets.

The option-length octet counts the option-type octet, the option-length octet, and the option-data octets.

The option-type octet is viewed as having three fields: a 1-bit copied flag, a 2-bit option class, and a 5-bit option number. These fields form an 8-bit value for the option type field. IP Options are commonly referred to by their 8-bit value.

For a complete list and description of IP Options, refer to RFC 791, *Internet Protocol* at the following URL: <http://www.faqs.org/rfcs/rfc791.html>

Benefits of Filtering IP Options

- Filtering of packets that contain IP Options from the network relieves downstream devices and hosts of the load from options packets.
- This feature also minimizes load to the Route Processor (RP) for packets with IP Options that require RP processing on distributed systems. Previously, the packets were always routed to or processed by the RP CPU. Filtering the packets prevents them from impacting the RP.

Benefits of Filtering on TCP Flags

The ACL TCP Flags Filtering feature provides a flexible mechanism for filtering on TCP flags. Previously, an incoming packet was matched as long as any TCP flag in the packet matched a flag specified in the access control entry (ACE). This behavior allows for a security loophole, because packets with all flags set could get past the access control list (ACL). The ACL TCP Flags Filtering feature allows you to select any combination of flags on which to filter. The ability to match on a flag set and on a flag not set gives you a greater degree of control for filtering on TCP flags, thus enhancing security.

Because TCP packets can be sent as false synchronization packets that can be accepted by a listening port, it is recommended that administrators of firewall devices set up some filtering rules to drop false TCP packets.

The ACEs that make up an access list can be configured to detect and drop unauthorized TCP packets by allowing only the packets that have a very specific group of TCP flags set or not set. The ACL TCP Flags Filtering feature provides a greater degree of packet-filtering control in the following ways:

- You can select any desired combination of TCP flags on which to filter TCP packets.
- You can configure ACEs to allow matching on a flag that is set, as well as on a flag that is not set.

TCP Flags

The table below lists the TCP flags, which are further described in RFC 793, *Transmission Control Protocol*.

Table 136: TCP Flags

TCP Flag	Purpose
ACK	Acknowledge flag—Indicates that the acknowledgment field of a segment specifies the next sequence number the sender of this segment is expecting to receive.
FIN	Finish flag—Used to clear connections.
PSH	Push flag—Indicates the data in the call should be immediately pushed through to the receiving user.
RST	Reset flag—Indicates that the receiver should delete the connection without further interaction.
SYN	Synchronize flag—Used to establish connections.
URG	Urgent flag—Indicates that the urgent field is meaningful and must be added to the segment sequence number.

How to Configure ACL Support for Filtering IP Options

Filtering Packets That Contain IP Options

Complete these steps to configure an access list to filter packets that contain IP options and to verify that the access list has been configured correctly.



Note

- The ACL Support for Filtering IP Options feature can be used only with named, extended ACLs.
- Resource Reservation Protocol (RSVP) Multiprotocol Label Switching Traffic Engineering (MPLS TE), Internet Group Management Protocol Version 2 (IGMPV2), and other protocols that use IP options packets may not function in drop or ignore mode if this feature is configured.
- On most Cisco devices, a packet with IP options is not switched in hardware, but requires control plane software processing (primarily because there is a need to process the options and rewrite the IP header), so all IP packets with IP options will be filtered and switched in software.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip access-list extended** *access-list-name*
4. [*sequence-number*] **deny** *protocol source source-wildcard destination destination-wildcard* [**option** *option-value*] [**precedence** *precedence*] [**tos** *tos*] [**log**] [**time-range** *time-range-name*] [**fragments**]
5. [*sequence-number*] **permit** *protocol source source-wildcard destination destination-wildcard* [**option** *option-value*] [**precedence** *precedence*] [**tos** *tos*] [**log**] [**time-range** *time-range-name*] [**fragments**]
6. Repeat Step 4 or Step 5 as necessary.

7. **end**
8. **show ip access-lists** *access-list-name*

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none">• Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	ip access-list extended <i>access-list-name</i> Example: Device(config)# ip access-list extended mylist1	Specifies the IP access list by name and enters named access list configuration mode.
Step 4	[<i>sequence-number</i>] deny <i>protocol source source-wildcard destination destination-wildcard</i> [option <i>option-value</i>] [precedence <i>precedence</i>] [tos <i>tos</i>] [log] [time-range <i>time-range-name</i>] [fragments] Example: Device(config-ext-nacl)# deny ip any any option traceroute	(Optional) Specifies a deny statement in named IP access list mode. <ul style="list-style-type: none">• This access list happens to use a deny statement first, but a permit statement could appear first, depending on the order of statements you need.• Use the option keyword and <i>option-value</i> argument to filter packets that contain a particular IP Option.• In this example, any packet that contains the traceroute IP option will be filtered out.• Use the no <i>sequence-number</i> form of this command to delete an entry.
Step 5	[<i>sequence-number</i>] permit <i>protocol source source-wildcard destination destination-wildcard</i> [option <i>option-value</i>] [precedence <i>precedence</i>] [tos <i>tos</i>] [log] [time-range <i>time-range-name</i>] [fragments] Example: Device(config-ext-nacl)# permit ip any any option security	Specifies a permit statement in named IP access list mode. <ul style="list-style-type: none">• In this example, any packet (not already filtered) that contains the security IP option will be permitted.• Use the no <i>sequence-number</i> form of this command to delete an entry.
Step 6	Repeat Step 4 or Step 5 as necessary.	Allows you to revise the access list.
Step 7	end Example: Device(config-ext-nacl)# end	(Optional) Exits named access list configuration mode and returns to privileged EXEC mode.
Step 8	show ip access-lists <i>access-list-name</i> Example:	(Optional) Displays the contents of the IP access list.

	Command or Action	Purpose
	Device# show ip access-lists mylist1	

Filtering Packets That Contain TCP Flags

This task configures an access list to filter packets that contain TCP flags and verifies that the access list has been configured correctly.



Note

- TCP flag filtering can be used only with named, extended ACLs.
- The ACL TCP Flags Filtering feature is supported only for Cisco ACLs.
- Previously, the following command-line interface (CLI) format could be used to configure a TCP flag-checking mechanism:

permit tcp any any rst The following format that represents the same access control entry (ACE) can now be used: **permit tcp any any match-any +rst** Both the CLI formats are accepted; however, if the new keywords **match-all** or **match-any** are chosen, they must be followed by the new flags that are prefixed with “+” or “-”. It is advisable to use only the old format or the new format in a single ACL. You cannot mix and match the old and new CLI formats.



Caution

If a device having ACEs with the new syntax format is reloaded with a previous version of the Cisco software that does not support the ACL TCP Flags Filtering feature, the ACEs will not be applied, leading to possible security loopholes.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip access-list extended** *access-list-name*
4. [*sequence-number*] **permit tcp** *source source-wildcard* [*operator* [*port*]] *destination destination-wildcard* [*operator* [*port*]] [**established** {**match-any** | **match-all**} {+ | -} *flag-name*] [**precedence** *precedence*] [**tos** *tos*] [**log**] [**time-range** *time-range-name*] [**fragments**]
5. [*sequence-number*] **deny tcp** *source source-wildcard* [*operator* [*port*]] *destination destination-wildcard* [*operator* [*port*]] [**established** {**match-any** | **match-all**} {+ | -} *flag-name*] [**precedence** *precedence*] [**tos** *tos*] [**log**] [**time-range** *time-range-name*] [**fragments**]
6. Repeat Step 4 or Step 5 as necessary, adding statements by sequence number where you planned. Use the **no** *sequence-number* command to delete an entry.
7. **end**
8. **show ip access-lists** *access-list-name*

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example:	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.

	Command or Action	Purpose
	Device> enable	
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	ip access-list extended <i>access-list-name</i> Example: Device(config)# ip access-list extended kmd1	Specifies the IP access list by name and enters named access list configuration mode.
Step 4	<i>[sequence-number]</i> permit tcp <i>source source-wildcard</i> <i>[operator [port]] destination destination-wildcard [operator [port]]</i> [established {match-any match-all} {+ -}] <i>flag-name</i> [precedence precedence] [tos tos] [log] [time-range time-range-name] [fragments] Example: Device(config-ext-nacl)# permit tcp any any match-any +rst	Specifies a permit statement in named IP access list mode. <ul style="list-style-type: none"> • This access list happens to use a permit statement first, but a deny statement could appear first, depending on the order of statements you need. • Use the TCP command syntax of the permit command. • Any packet with the RST TCP header flag set will be matched and allowed to pass the named access list kmd1 in Step 3.
Step 5	<i>[sequence-number]</i> deny tcp <i>source source-wildcard</i> <i>[operator [port]] destination destination-wildcard [operator [port]]</i> [established {match-any match-all} {+ -}] <i>flag-name</i> [precedence precedence] [tos tos] [log] [time-range time-range-name] [fragments] Example: Device(config-ext-nacl)# deny tcp any any match-all -ack -fin	(Optional) Specifies a deny statement in named IP access list mode. <ul style="list-style-type: none"> • This access list happens to use a permit statement first, but a deny statement could appear first, depending on the order of statements you need. • Use the TCP command syntax of the deny command. • Any packet that does not have the ACK flag set, and also does not have the FIN flag set, will not be allowed to pass the named access list kmd1 in Step 3. • See the deny(IP) command for additional command syntax to permit upper-layer protocols (ICMP, IGMP, TCP, and UDP).
Step 6	Repeat Step 4 or Step 5 as necessary, adding statements by sequence number where you planned. Use the no sequence-number command to delete an entry.	Allows you to revise the access list.
Step 7	end Example: Device(config-ext-nacl)# end	(Optional) Exits the configuration mode and returns to privileged EXEC mode.
Step 8	show ip access-lists <i>access-list-name</i>	(Optional) Displays the contents of the IP access list.

	Command or Action	Purpose
	Example: Device# show ip access-lists kmdl	<ul style="list-style-type: none"> Review the output to confirm that the access list includes the new entry.

Configuration Examples for ACL Support for Filtering IP Options

Example: Filtering Packets That Contain IP Options

The following example shows an extended access list named mylist2 that contains access list entries (ACEs) that are configured to permit TCP packets only if they contain the IP Options that are specified in the ACEs:

```
ip access-list extended mylist2
 10 permit ip any any option eool
 20 permit ip any any option record-route
 30 permit ip any any option zsu
 40 permit ip any any option mtup
```

The **show access-list** command has been entered to show how many packets were matched and therefore permitted:

```
Device# show ip access-list mylist2
Extended IP access list test
10 permit ip any any option eool (1 match)
20 permit ip any any option record-route (1 match)
30 permit ip any any option zsu (1 match)
40 permit ip any any option mtup (1 match)
```

Example: Filtering Packets That Contain TCP Flags

The following access list allows TCP packets only if the TCP flags ACK and SYN are set and the FIN flag is not set:

```
ip access-list extended aaa
 permit tcp any any match-all +ack +syn -fin
end
```

The **show access-list** command has been entered to display the ACL:

```
Device# show access-list aaa

Extended IP access list aaa
 10 permit tcp any any match-all +ack +syn -fin
```

Additional References for ACL Support for Filtering IP Options

Related Documents

Related Topic	Document Title
Cisco security commands	<ul style="list-style-type: none"> • Cisco IOS Security Command Reference: Commands A to C • Cisco IOS Security Command Reference: Commands D to L • Cisco IOS Security Command Reference: Commands M to R • Cisco IOS Security Command Reference: Commands S to Z

RFCs

RFC	Title
RFC 791	<i>Internet Protocol</i> http://www.faqs.org/rfcs/rfc791.html
RFC 793	<i>Transmission Control Protocol</i>
RFC 1393	<i>Traceroute Using an IP Option</i>

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

Feature Information for Creating an IP Access List to Filter

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Table 137: Feature Information for Creating an IP Access List to Filter

Feature Name	Releases	Feature Configuration Information
ACL Support for Filtering IP Options	Cisco IOS 15.2(2)E	This feature allows you to filter packets having IP Options, in order to prevent routers from becoming saturated with spurious packets.
ACL TCP Flags Filtering	Cisco IOS 15.2(2)E	This feature provides a flexible mechanism for filtering on TCP flags. The ACL TCP Flags Filtering feature allows you to select any combination of flags on which to filter. The ability to match on a flag set and on a flag not set gives you a greater degree of control for filtering on TCP flags, thus enhancing security.



CHAPTER 70

VLAN Access Control Lists

VLAN access control lists (ACLs) or VLAN maps access-control all packets (bridged and routed). You can use VLAN maps to filter traffic between devices in the same VLAN. VLAN maps are configured to provide access control based on Layer 3 addresses for IPv4. Unsupported protocols are access-controlled through MAC addresses using Ethernet access control entries (ACEs). After a VLAN map is applied to a VLAN, all packets (routed or bridged) entering the VLAN are checked against the VLAN map. Packets can either enter the VLAN through a switch port or through a routed port after being routed.

This module provides more information about VLAN ACLs and how to configure them.

- [Information About VLAN Access Control Lists, on page 1377](#)
- [How to Configure VLAN Access Control Lists, on page 1380](#)
- [Configuration Examples for ACLs and VLAN Maps, on page 1389](#)
- [Configuration Examples for Using VLAN Maps in Your Network, on page 1391](#)
- [Configuration Examples of Router ACLs and VLAN Maps Applied to VLANs, on page 1393](#)

Information About VLAN Access Control Lists

VLAN Maps

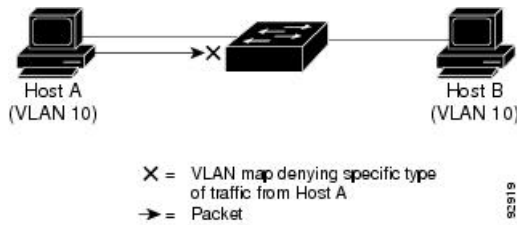
VLAN ACLs or VLAN maps are used to control network traffic within a VLAN. You can apply VLAN maps to all packets that are bridged within a VLAN in the switch or switch stack. VACLs are strictly for security packet filtering and for redirecting traffic to specific physical interfaces. VACLs are not defined by direction (ingress or egress).

All non-IP protocols are access-controlled through MAC addresses and Ethertype using MAC VLAN maps. (IP traffic is not access controlled by MAC VLAN maps.) You can enforce VLAN maps only on packets going through the switch; you cannot enforce VLAN maps on traffic between hosts on a hub or on another switch connected to this switch.

With VLAN maps, forwarding of packets is permitted or denied, based on the action specified in the map.

Figure 97: Using VLAN Maps to Control Traffic

This shows how a VLAN map is applied to prevent a specific type of traffic from Host A in VLAN 10 from being forwarded. You can apply only one VLAN map to a VLAN.



VLAN Map Configuration Guidelines

VLAN maps are the only way to control filtering within a VLAN. VLAN maps have no direction. To filter traffic in a specific direction by using a VLAN map, you need to include an ACL with specific source or destination addresses. If there is a match clause for that type of packet (IP or MAC) in the VLAN map, the default action is to drop the packet if the packet does not match any of the entries within the map. If there is no match clause for that type of packet, the default is to forward the packet.

The following are the VLAN map configuration guidelines:

- If there is no ACL configured to deny traffic on an interface and no VLAN map is configured, all traffic is permitted.
- Each VLAN map consists of a series of entries. The order of entries in an VLAN map is important. A packet that comes into the switch is tested against the first entry in the VLAN map. If it matches, the action specified for that part of the VLAN map is taken. If there is no match, the packet is tested against the next entry in the map.
- If the VLAN map has at least one match clause for the type of packet (IP or MAC) and the packet does not match any of these match clauses, the default is to drop the packet. If there is no match clause for that type of packet in the VLAN map, the default is to forward the packet.
- Logging is not supported for VLAN maps.
- When a switch has an IP access list or MAC access list applied to a Layer 2 interface, and you apply a VLAN map to a VLAN that the port belongs to, the port ACL takes precedence over the VLAN map.
- If a VLAN map configuration cannot be applied in hardware, all packets in that VLAN are dropped.

VLAN Maps with Router ACLs

To access control both bridged and routed traffic, you can use VLAN maps only or a combination of router ACLs and VLAN maps. You can define router ACLs on both input and output routed VLAN interfaces, and you can define a VLAN map to access control the bridged traffic.

If a packet flow matches a VLAN-map deny clause in the ACL, regardless of the router ACL configuration, the packet flow is denied.



Note When you use router ACLs with VLAN maps, packets that require logging on the router ACLs are not logged if they are denied by a VLAN map.

If the VLAN map has a match clause for the type of packet (IP or MAC) and the packet does not match the type, the default is to drop the packet. If there is no match clause in the VLAN map, and no action specified, the packet is forwarded if it does not match any VLAN map entry.

VLAN Maps and Router ACL Configuration Guidelines

These guidelines are for configurations where you need to have an router ACL and a VLAN map on the same VLAN. These guidelines do not apply to configurations where you are mapping router ACLs and VLAN maps on different VLANs.

If you must configure a router ACL and a VLAN map on the same VLAN, use these guidelines for both router ACL and VLAN map configuration:

- You can configure only one VLAN map and one router ACL in each direction (input/output) on a VLAN interface.
- Whenever possible, try to write the ACL with all entries having a single action except for the final, default action of the other type. That is, write the ACL using one of these two forms:

```
permit... permit... permit... deny ip any any
```

or

```
deny... deny... deny... permit ip any any
```
- To define multiple actions in an ACL (permit, deny), group each action type together to reduce the number of entries.
- Avoid including Layer 4 information in an ACL; adding this information complicates the merging process. The best merge results are obtained if the ACLs are filtered based on IP addresses (source and destination) and not on the full flow (source IP address, destination IP address, protocol, and protocol ports). It is also helpful to use *don't care* bits in the IP address, whenever possible.

If you need to specify the full-flow mode and the ACL contains both IP ACEs and TCP/UDP/ICMP ACEs with Layer 4 information, put the Layer 4 ACEs at the end of the list. This gives priority to the filtering of traffic based on IP addresses.

VACL Logging

When you configure VACL logging, syslog messages are generated for denied IP packets under these circumstances:

- When the first matching packet is received.
- For any matching packets received within the last 5 minutes.
- If the threshold is reached before the 5-minute interval.

Log messages are generated on a per-flow basis. A flow is defined as packets with the same IP addresses and Layer 4 (UDP or TCP) port numbers. If a flow does not receive any packets in the 5-minute interval, that flow is removed from the cache. When a syslog message is generated, the timer and packet counter are reset.

VACL logging restrictions:

- Only denied IP packets are logged.
- Packets that require logging on the outbound port ACLs are not logged if they are denied by a VACL.

How to Configure VLAN Access Control Lists

Creating Named MAC Extended ACLs

You can filter non-IPv4 traffic on a VLAN or on a Layer 2 interface by using MAC addresses and named MAC extended ACLs. The procedure is similar to that of configuring other extended named ACLs.

Follow these steps to create a named MAC extended ACL:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **mac access-list extended *name***
4. **{deny | permit} {any | host *source MAC address* | *source MAC address mask*} {any | host *destination MAC address* | *destination MAC address mask*} [*type mask* | **lsap lsap mask** | **aarp amber** | **dec-spanning** | **decnet-iv** | **diagnostic dsm** | **etype-6000 etype-8042** | **lat** | **lave-sca** | **mop-console** | **mop-dump** | **msdos** | **mumps** | **netbios** | **vines-echo** | **vines-ip** | **xns-idp** | 0-65535] [**cos** *cos*]**
5. **end**
6. **show running-config**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.

	Command or Action	Purpose
Step 3	<p>mac access-list extended <i>name</i></p> <p>Example:</p> <pre>Device(config)# mac access-list extended macl</pre>	Defines an extended MAC access list using a name.
Step 4	<p>{deny permit} {any host <i>source MAC address</i> <i>source MAC address mask</i>} {any host <i>destination MAC address</i> <i>destination MAC address mask</i>} [<i>type mask</i> lsap <i>lsap mask</i> aarp amber dec-spanning decnet-iv diagnostic dsm etype-6000 etype-8042 lat lavr-sca mop-console mop-dump msdos mumps netbios vines-echo vines-ip xns-idp 0-65535] [cos <i>cos</i>]</p> <p>Example:</p> <pre>Device(config-ext-macl)# deny any any decnet-iv</pre> <p>or</p> <pre>Device(config-ext-macl)# permit any any</pre>	<p>In extended MAC access-list configuration mode, specifies to permit or deny any source MAC address, a source MAC address with a mask, or a specific host source MAC address and any destination MAC address, destination MAC address with a mask, or a specific destination MAC address.</p> <p>(Optional) You can also enter these options:</p> <ul style="list-style-type: none"> • type mask—An arbitrary EtherType number of a packet with Ethernet II or SNAP encapsulation in decimal, hexadecimal, or octal with optional mask of <i>don't care</i> bits applied to the EtherType before testing for a match. • lsap lsap mask—An LSAP number of a packet with IEEE 802.2 encapsulation in decimal, hexadecimal, or octal with optional mask of <i>don't care</i> bits. • aarp amber dec-spanning decnet-iv diagnostic dsm etype-6000 etype-8042 lat lavr-sca mop-console mop-dump msdos mumps netbios vines-echo vines-ip xns-idp—A non-IP protocol. • cos cos—An IEEE 802.1Q cost of service number from 0 to 7 used to set priority.
Step 5	<p>end</p> <p>Example:</p> <pre>Device(config-ext-macl)# end</pre>	Returns to privileged EXEC mode.
Step 6	<p>show running-config</p> <p>Example:</p> <pre>Device# show running-config</pre>	Verifies your entries.
Step 7	<p>copy running-config startup-config</p> <p>Example:</p> <pre>Device# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Applying a MAC ACL to a Layer 2 Interface

Follow these steps to apply a MAC access list to control access to a Layer 2 interface:

SUMMARY STEPS

1. **configure terminal**
2. **configure terminal**
3. **interface *interface-id***
4. **mac access-group {*name*} {in | out }**
5. **end**
6. **show mac access-group [interface *interface-id*]**
7. **configure terminal**
8. **configure terminal**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	interface <i>interface-id</i> Example: Device(config)# interface gigabitethernet1/0/2	Identifies a specific interface, and enter interface configuration mode. The interface must be a physical Layer 2 interface (port ACL).
Step 4	mac access-group {<i>name</i>} {in out } Example: Device(config-if)# mac access-group mac1 in	Controls access to the specified interface by using the MAC access list. Port ACLs are supported in the outbound and inbound directions .
Step 5	end Example: Device(config-if)# end	Returns to privileged EXEC mode.

	Command or Action	Purpose
Step 6	show mac access-group [<i>interface interface-id</i>] Example: Device# show mac access-group interface gigabitethernet1/0/2	Displays the MAC access list applied to the interface or all Layer 2 interfaces.
Step 7	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 8	configure terminal Example: Device# configure terminal	Enters global configuration mode.

After receiving a packet, the switch checks it against the inbound ACL. If the ACL permits it, the switch continues to process the packet. If the ACL rejects the packet, the switch discards it. When you apply an undefined ACL to an interface, the switch acts as if the ACL has not been applied and permits all packets. Remember this behavior if you use undefined ACLs for network security.

Configuring VLAN Maps

To create a VLAN map and apply it to one or more VLANs, perform these steps:

Before you begin

Create the standard or extended IPv4 ACLs or named MAC extended ACLs that you want to apply to the VLAN.

SUMMARY STEPS

1. **vlan access-map** *name* [**number**]
2. **match** {**ip** | **mac**} **address** {*name* | *number*} [*name* | *number*]
3. Enter one of the following commands to specify an IP packet or a non-IP packet (with only a known MAC address) and to match the packet against one or more ACLs (standard or extended):

- **action** { **forward** }

```
Device(config-access-map)# action forward
```

- **action** { **drop** }

```
Device(config-access-map)# action drop
```

4. `vlan filter mapname vlan-list list`

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p><code>vlan access-map name [number]</code></p> <p>Example:</p> <pre>Device(config)# vlan access-map map_1 20</pre>	<p>Creates a VLAN map, and give it a name and (optionally) a number. The number is the sequence number of the entry within the map.</p> <p>When you create VLAN maps with the same name, numbers are assigned sequentially in increments of 10. When modifying or deleting maps, you can enter the number of the map entry that you want to modify or delete.</p> <p>VLAN maps do not use the specific permit or deny keywords. To deny a packet by using VLAN maps, create an ACL that would match the packet, and set the action to drop. A permit in the ACL counts as a match. A deny in the ACL means no match.</p> <p>Entering this command changes to access-map configuration mode.</p>
Step 2	<p><code>match {ip mac} address {name number} [name number]</code></p> <p>Example:</p> <pre>Device(config-access-map)# match ip address ip2</pre>	<p>Match the packet (using either the IP or MAC address) against one or more standard or extended access lists. Note that packets are only matched against access lists of the correct protocol type. IP packets are matched against standard or extended IP access lists. Non-IP packets are only matched against named MAC extended access lists.</p> <p>Note If the VLAN map is configured with a match clause for a type of packet (IP or MAC) and the map action is drop, all packets that match the type are dropped. If the VLAN map has no match clause, and the configured action is drop, all IP and Layer 2 packets are dropped.</p>
Step 3	<p>Enter one of the following commands to specify an IP packet or a non-IP packet (with only a known MAC address) and to match the packet against one or more ACLs (standard or extended):</p> <ul style="list-style-type: none"> • <code>action { forward }</code> <pre>Device(config-access-map)# action forward</pre> <ul style="list-style-type: none"> • <code>action { drop }</code> <pre>Device(config-access-map)# action drop</pre>	<p>Sets the action for the map entry.</p>
Step 4	<p><code>vlan filter mapname vlan-list list</code></p>	<p>Applies the VLAN map to one or more VLAN IDs.</p>

	Command or Action	Purpose
	Example: Device(config)# vlan filter map 1 vlan-list 20-22	The list can be a single VLAN ID (22), a consecutive list (10-22), or a string of VLAN IDs (12, 22, 30). Spaces around the comma and hyphen are optional.

Creating a VLAN Map

Each VLAN map consists of an ordered series of entries. Beginning in privileged EXEC mode, follow these steps to create, add to, or delete a VLAN map entry:

SUMMARY STEPS

1. **configure terminal**
2. **vlan access-map** *name* [**number**]
3. **match** {**ip** | **mac**} **address** {*name* | *number*} [*name* | *number*]
4. **action** {**drop** | **forward**}
5. **end**
6. **show running-config**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 2	vlan access-map <i>name</i> [number] Example: Device(config)# vlan access-map map_1 20	<p>Creates a VLAN map, and give it a name and (optionally) a number. The number is the sequence number of the entry within the map.</p> <p>When you create VLAN maps with the same name, numbers are assigned sequentially in increments of 10. When modifying or deleting maps, you can enter the number of the map entry that you want to modify or delete.</p> <p>VLAN maps do not use the specific permit or deny keywords. To deny a packet by using VLAN maps, create an ACL that would match the packet, and set the action to drop. A permit in the ACL counts as a match. A deny in the ACL means no match.</p> <p>Entering this command changes to access-map configuration mode.</p>
Step 3	match { ip mac } address { <i>name</i> <i>number</i> } [<i>name</i> <i>number</i>]	Match the packet (using either the IP or MAC address) against one or more standard or extended access lists. Note

	Command or Action	Purpose
	Example: Device(config-access-map) # match ip address ip2	that packets are only matched against access lists of the correct protocol type. IP packets are matched against standard or extended IP access lists. Non-IP packets are only matched against named MAC extended access lists.
Step 4	action {drop forward} Example: Device(config-access-map) # action forward	(Optional) Sets the action for the map entry. The default is to forward.
Step 5	end Example: Device(config-access-map) # end	Returns to global configuration mode.
Step 6	show running-config Example: Device# show running-config	Displays the access list configuration.
Step 7	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Applying a VLAN Map to a VLAN

Beginning in privileged EXEC mode, follow these steps to apply a VLAN map to one or more VLANs:

SUMMARY STEPS

- 1.
2. **configure terminal**
3. **vlan filter *mapname* vlan-list *list***
4. **end**
5. **show running-config**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1		

	Command or Action	Purpose
Step 2	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode.
Step 3	vlan filter <i>mapname</i> vlan-list <i>list</i> Example: Device(config)# <code>vlan filter map 1 vlan-list 20-22</code>	Applies the VLAN map to one or more VLAN IDs. The list can be a single VLAN ID (22), a consecutive list (10-22), or a string of VLAN IDs (12, 22, 30). Spaces around the comma and hyphen are optional.
Step 4	end Example: Device(config)# <code>end</code>	Returns to privileged EXEC mode.
Step 5	show running-config Example: Device# <code>show running-config</code>	Displays the access list configuration.
Step 6	copy running-config startup-config Example: Device# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Configuring VACL Logging

Beginning in privileged EXEC mode:

SUMMARY STEPS

1. `configure terminal`
2. `vlan access-map name [number]`
3. `action drop log`
4. `exit`
5. `vlan access-log {maxflow max_number | threshold pkt_count}`
6. `end`

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 2	vlan access-map name [number] Example: Device(config) # vlan access-map gandymede 10	<p>Creates a VLAN map. Give it a name and optionally a number. The number is the sequence number of the entry within the map.</p> <p>The sequence number range is from 0 to 65535.</p> <p>When you create VLAN maps with the same name, numbers are assigned sequentially in increments of 10. When modifying or deleting maps, you can enter the number of the map entry that you want to modify or delete.</p> <p>Specifying the map name and optionally a number enters the access-map configuration mode.</p>
Step 3	action drop log Example: Device(config-access-map) # action drop log	Sets the VLAN access map to drop and log IP packets.
Step 4	exit Example: Device(config-access-map) # exit	Exits the VLAN access map configuration mode and return to the global configuration mode.
Step 5	vlan access-log {maxflow max_number threshold pkt_count} Example: Device(config) # vlan access-log threshold 4000	<p>Configures the VACL logging parameters.</p> <ul style="list-style-type: none"> • maxflow max_number—Sets the log table size. The content of the log table can be deleted by setting the maxflow to 0. When the log table is full, the software drops logged packets from new flows. The range is from 0 to 2048. The default is 500. • threshold pkt_count—Sets the logging threshold. A logging message is generated if the threshold for a flow is reached before the 5-minute interval. The threshold range is from 0 to 2147483647. The default threshold is 0, which means that a syslog message is generated every 5 minutes.
Step 6	end Example:	Returns to privileged EXEC mode.

	Command or Action	Purpose
	Device(config)# end	

Configuration Examples for ACLs and VLAN Maps

Example: Creating an ACL and a VLAN Map to Deny a Packet

This example shows how to create an ACL and a VLAN map to deny a packet. In the first map, any packets that match the *ip1* ACL (TCP packets) would be dropped. You first create the *ip1* ACL to permit any TCP packet and no other packets. Because there is a match clause for IP packets in the VLAN map, the default action is to drop any IP packet that does not match any of the match clauses.

```
Device(config)# ip access-list extended ip1
Device(config-ext-nacl)# permit tcp any any
Device(config-ext-nacl)# exit
Device(config)# vlan access-map map_1 10
Device(config-access-map)# match ip address ip1
Device(config-access-map)# action drop
```

Example: Creating an ACL and a VLAN Map to Permit a Packet

This example shows how to create a VLAN map to permit a packet. ACL *ip2* permits UDP packets and any packets that match the *ip2* ACL are forwarded. In this map, any IP packets that did not match any of the previous ACLs (that is, packets that are not TCP packets or UDP packets) would get dropped.

```
Device(config)# ip access-list extended ip2
Device(config-ext-nacl)# permit udp any any
Device(config-ext-nacl)# exit
Device(config)# vlan access-map map_1 20
Device(config-access-map)# match ip address ip2
Device(config-access-map)# action forward
```

Example: Default Action of Dropping IP Packets and Forwarding MAC Packets

In this example, the VLAN map has a default action of drop for IP packets and a default action of forward for MAC packets. Used with standard ACL 101 and extended named access lists **igmp-match** and **tcp-match**, the map will have the following results:

- Forward all UDP packets
- Drop all IGMP packets
- Forward all TCP packets
- Drop all other IP packets
- Forward all non-IP packets

```

Device(config)# access-list 101 permit udp any any
Device(config)# ip access-list extended igmp-match
Device(config-ext-nacl)# permit igmp any any

Device(config-ext-nacl)# permit tcp any any
Device(config-ext-nacl)# exit
Device(config)# vlan access-map drop-ip-default 10
Device(config-access-map)# match ip address 101
Device(config-access-map)# action forward
Device(config-access-map)# exit
Device(config)# vlan access-map drop-ip-default 20
Device(config-access-map)# match ip address igmp-match
Device(config-access-map)# action drop
Device(config-access-map)# exit
Device(config)# vlan access-map drop-ip-default 30
Device(config-access-map)# match ip address tcp-match
Device(config-access-map)# action forward

```

Example: Default Action of Dropping MAC Packets and Forwarding IP Packets

In this example, the VLAN map has a default action of drop for MAC packets and a default action of forward for IP packets. Used with MAC extended access lists **good-hosts** and **good-protocols**, the map will have the following results:

- Forward MAC packets from hosts 0000.0c00.0111 and 0000.0c00.0211
- Forward MAC packets with decnet-iv or vines-ip protocols
- Drop all other non-IP packets
- Forward all IP packets

Example: Default Action of Dropping All Packets

In this example, the VLAN map has a default action of drop for all packets (IP and non-IP). Used with access lists **tcp-match** and **good-hosts** from Examples 2 and 3, the map will have the following results:

- Forward all TCP packets
- Forward MAC packets from hosts 0000.0c00.0111 and 0000.0c00.0211
- Drop all other IP packets
- Drop all other MAC packets

```

Device(config)# vlan access-map drop-all-default 10
Device(config-access-map)# match ip address tcp-match
Device(config-access-map)# action forward
Device(config-access-map)# exit
Device(config)# vlan access-map drop-all-default 20
Device(config-access-map)# match mac address good-hosts
Device(config-access-map)# action forward

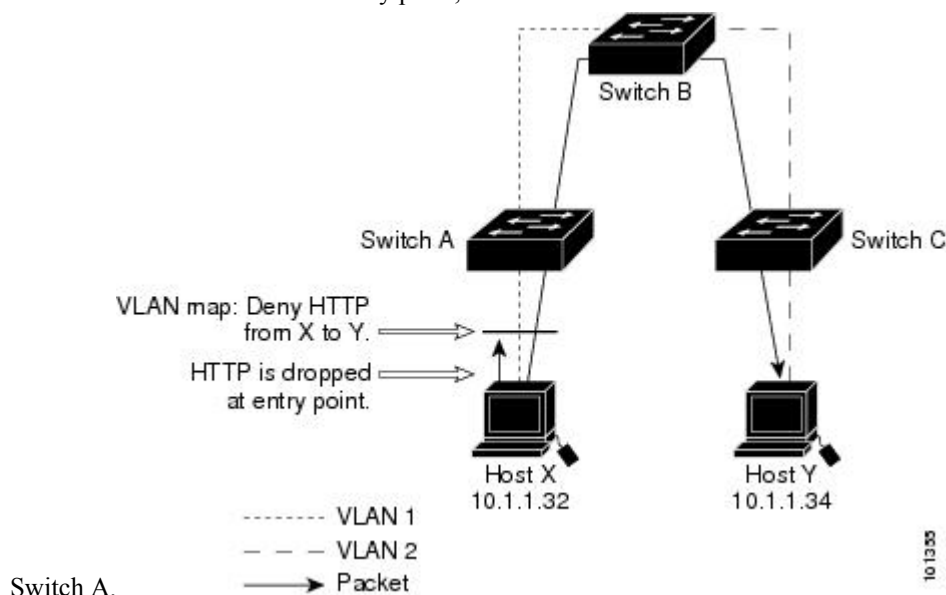
```


Configuration Examples for Using VLAN Maps in Your Network

Example: Wiring Closet Configuration

Figure 98: Wiring Closet Configuration

In a wiring closet configuration, routing might not be enabled on the switch. In this configuration, the switch can still support a VLAN map and a QoS classification ACL. Assume that Host X and Host Y are in different VLANs and are connected to wiring closet switches A and C. Traffic from Host X to Host Y is eventually being routed by Switch B, a Layer 3 switch with routing enabled. Traffic from Host X to Host Y can be access-controlled at the traffic entry point,



If you do not want HTTP traffic switched from Host X to Host Y, you can configure a VLAN map on Switch A to drop all HTTP traffic from Host X (IP address 10.1.1.32) to Host Y (IP address 10.1.1.34) at Switch A and not bridge it to Switch B.

First, define the IP access list *http* that permits (matches) any TCP traffic on the HTTP port.

```
Device(config)# ip access-list extended http
Device(config-ext-nacl)# permit tcp host 10.1.1.32 host 10.1.1.34 eq www
Device(config-ext-nacl)# exit
```

Next, create VLAN access map *map2* so that traffic that matches the *http* access list is dropped and all other IP traffic is forwarded.

```
Device(config)# vlan access-map map2 10
Device(config-access-map)# match ip address http
Device(config-access-map)# action drop
Device(config-access-map)# exit
Device(config)# ip access-list extended match_all
Device(config-ext-nacl)# permit ip any any
Device(config-ext-nacl)# exit
```

```
Device(config)# vlan access-map map2 20
Device(config-access-map)# match ip address match_all
Device(config-access-map)# action forward
```

Then, apply VLAN access map *map2* to VLAN 1.

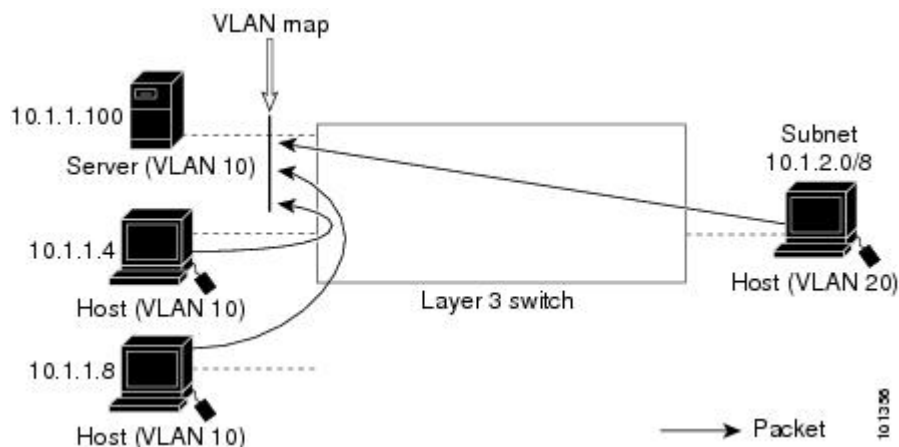
```
Device(config)# vlan filter map2 vlan 1
```

Example: Restricting Access to a Server on Another VLAN

Figure 99: Restricting Access to a Server on Another VLAN

You can restrict access to a server on another VLAN. For example, server 10.1.1.100 in VLAN 10 needs to have access denied to these hosts:

- Hosts in subnet 10.1.2.0/8 in VLAN 20 should not have access.
- Hosts 10.1.1.4 and 10.1.1.8 in VLAN 10 should not have access.



Example: Denying Access to a Server on Another VLAN

This example shows how to deny access to a server on another VLAN by creating the VLAN map SERVER1 that denies access to hosts in subnet 10.1.2.0.8, host 10.1.1.4, and host 10.1.1.8 and permits other IP traffic. The final step is to apply the map SERVER1 to VLAN 10.

Define the IP ACL that will match the correct packets.

```
Device(config)# ip access-list extended SERVER1_ACL
Device(config-ext-nacl)# permit ip 10.1.2.0 0.0.0.255 host 10.1.1.100
Device(config-ext-nacl)# permit ip host 10.1.1.4 host 10.1.1.100
Device(config-ext-nacl)# permit ip host 10.1.1.8 host 10.1.1.100
Device(config-ext-nacl)# exit
```

Define a VLAN map using this ACL that will drop IP packets that match SERVER1_ACL and forward IP packets that do not match the ACL.

```

Device(config)# vlan access-map SERVER1_MAP
Device(config-access-map)# match ip address SERVER1_ACL
Device(config-access-map)# action drop
Device(config)# vlan access-map SERVER1_MAP 20
Device(config-access-map)# action forward
Device(config-access-map)# exit

```

Apply the VLAN map to VLAN 10.

```

Device(config)# vlan filter SERVER1_MAP vlan-list 10

```

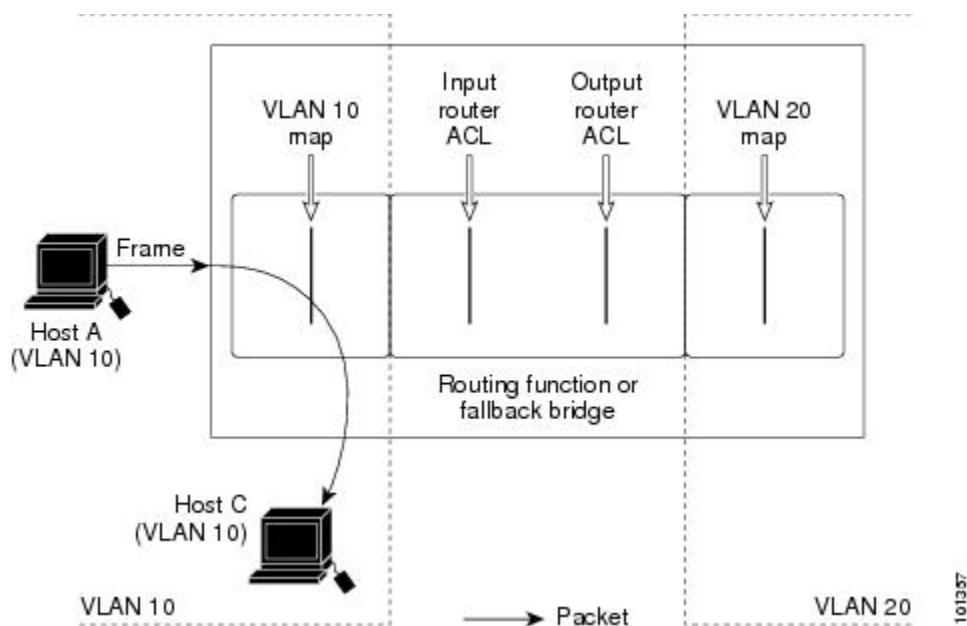
Configuration Examples of Router ACLs and VLAN Maps Applied to VLANs

This section gives examples of applying router ACLs and VLAN maps to a VLAN for switched, bridged, routed, and multicast packets. Although the following illustrations show packets being forwarded to their destination, each time the packet's path crosses a line indicating a VLAN map or an ACL, it is also possible that the packet might be dropped, rather than forwarded.

Example: ACLs and Switched Packets

Figure 100: Applying ACLs on Switched Packets

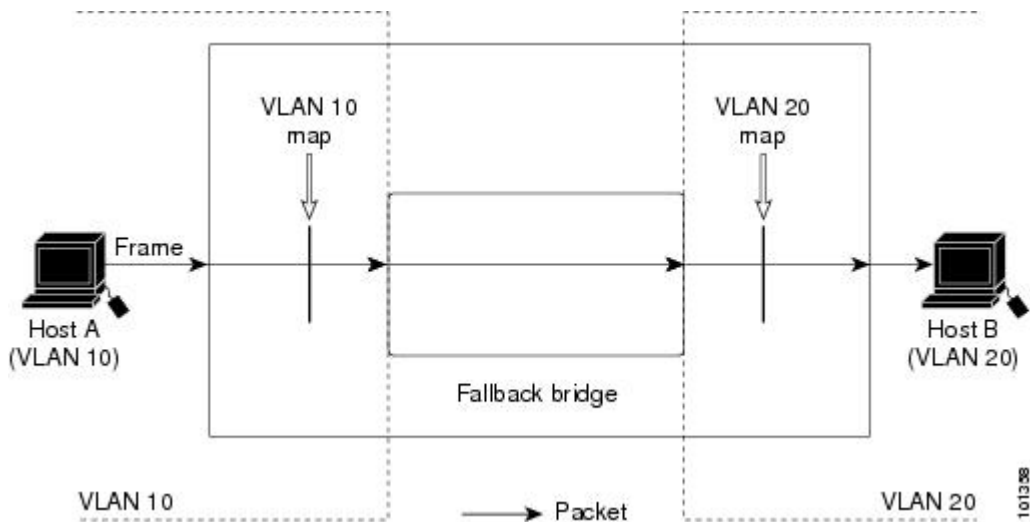
This example shows how an ACL is applied on packets that are switched within a VLAN. Packets switched within the VLAN without being routed or forwarded by fallback bridging are only subject to the VLAN map of the input VLAN.



Example: ACLs and Bridged Packets

Figure 101: Applying ACLs on Bridged Packets

This example shows how an ACL is applied on fallback-bridged packets. For bridged packets, only Layer 2 ACLs are applied to the input VLAN. Only non-IP, non-ARP packets can be fallback-bridged.

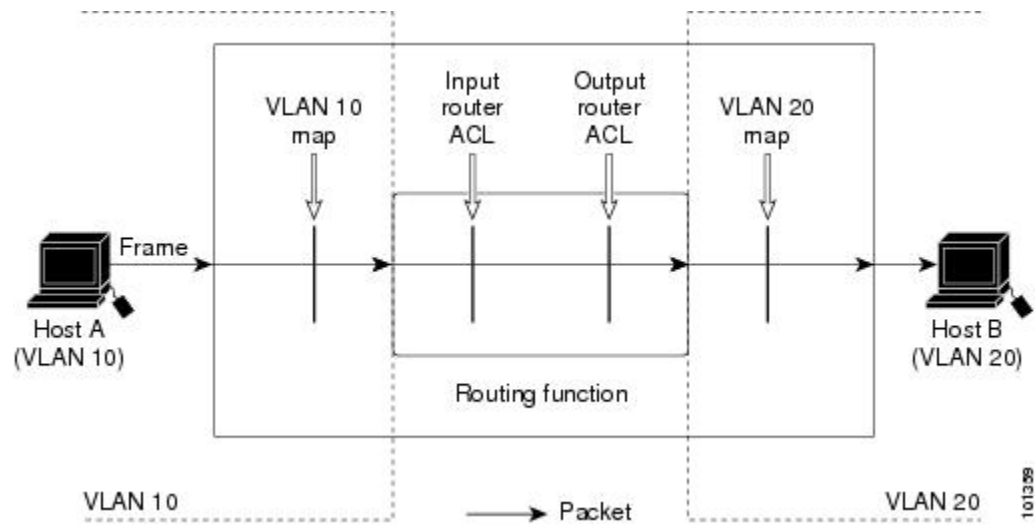


Example: ACLs and Routed Packets

Figure 102: Applying ACLs on Routed Packets

This example shows how ACLs are applied on routed packets. The ACLs are applied in this order:

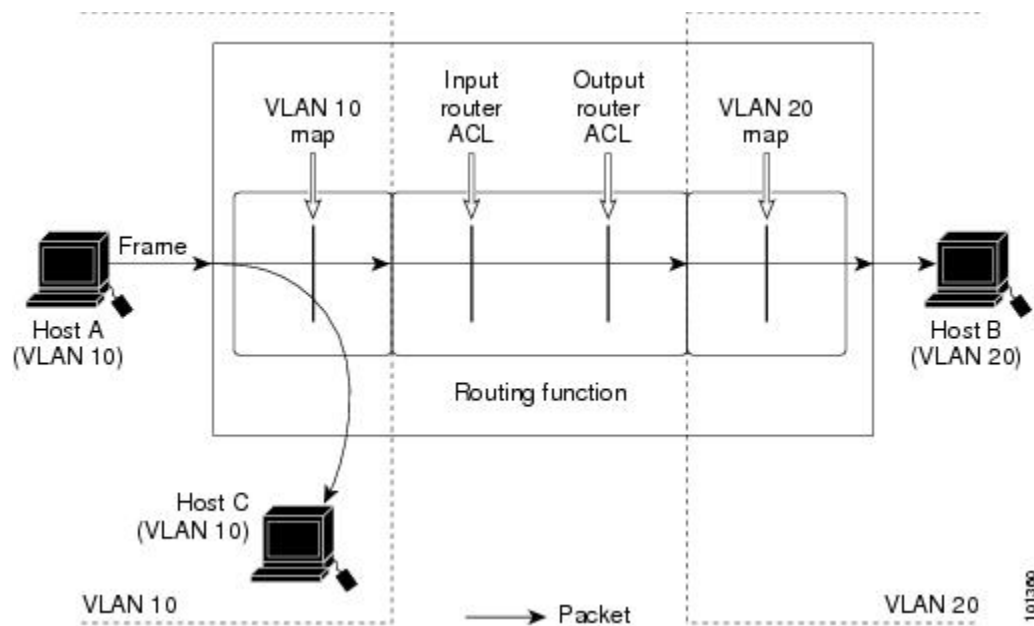
1. VLAN map for input VLAN
2. Input router ACL
3. Output router ACL
4. VLAN map for output VLAN



Example: ACLs and Multicast Packets

Figure 103: Applying ACLs on Multicast Packets

This example shows how ACLs are applied on packets that are replicated for IP multicasting. A multicast packet being routed has two different kinds of filters applied: one for destinations that are other ports in the input VLAN and another for each of the destinations that are in other VLANs to which the packet has been routed. The packet might be routed to more than one output VLAN, in which case a different router output ACL and VLAN map would apply for each destination VLAN. The final result is that the packet might be permitted in some of the output VLANs and not in others. A copy of the packet is forwarded to those destinations where it is permitted. However, if the input VLAN map drops the packet, no destination receives a copy of the packet.





CHAPTER 71

Configuring DHCP

- [Restrictions for DHCP, on page 1397](#)
- [Information About DHCP, on page 1397](#)
- [How to Configure DHCP Features, on page 1404](#)
- [Configuring DHCP Server Port-Based Address Allocation, on page 1413](#)

Restrictions for DHCP

The following scenario is not supported:

A non-DHCP snooping VLAN, and the SVI of the non-DHCP snooping VLAN is configured on a device. The SVI of the non-DHCP snooping VLAN is configured with the status of *no shutdown*. In this scenario, the DHCP packets in the non-DHCP snooping VLAN are not forwarded to the trusted ports.

If the SVI of the non-DHCP snooping VLAN is not configured or is configured with the *shutdown* status, DHCP packets are forwarded to the trusted ports, and DHCP clients can obtain IP address from the DHCP server.

Information About DHCP

DHCP Server

The DHCP server assigns IP addresses from specified address pools on a switch or router to DHCP clients and manages them. If the DHCP server cannot give the DHCP client the requested configuration parameters from its database, it forwards the request to one or more secondary DHCP servers defined by the network administrator. The switch can act as a DHCP server.

DHCP Relay Agent

A DHCP relay agent is a Layer 3 device that forwards DHCP packets between clients and servers. Relay agents forward requests and replies between clients and servers when they are not on the same physical subnet. Relay agent forwarding is different from the normal Layer 2 forwarding, in which IP datagrams are switched transparently between networks. Relay agents receive DHCP messages and generate new DHCP messages to send on output interfaces.

DHCP Snooping

DHCP snooping is a DHCP security feature that provides network security by filtering untrusted DHCP messages and by building and maintaining a DHCP snooping binding database, also referred to as a DHCP snooping binding table.

DHCP snooping acts like a firewall between untrusted hosts and DHCP servers. You use DHCP snooping to differentiate between untrusted interfaces connected to the end user and trusted interfaces connected to the DHCP server or another switch.



Note For DHCP snooping to function properly, all DHCP servers must be connected to the switch through trusted interfaces.

An untrusted DHCP message is a message that is received through an untrusted interface. By default, the switch considers all interfaces untrusted. So, the switch must be configured to trust some interfaces to use DHCP Snooping. When you use DHCP snooping in a service-provider environment, an untrusted message is sent from a device that is not in the service-provider network, such as a customer's switch. Messages from unknown devices are untrusted because they can be sources of traffic attacks.

The DHCP snooping binding database has the MAC address, the IP address, the lease time, the binding type, the VLAN number, and the interface information that corresponds to the local untrusted interfaces of a switch. It does not have information regarding hosts interconnected with a trusted interface.



Note When configuring DHCP snooping to block unauthorized IP address using the **ip verify source prot-security** command on an interface, the **switchport port-security** command should also be configured.

In a service-provider network, an example of an interface you might configure as trusted is one connected to a port on a device in the same network. An example of an untrusted interface is one that is connected to an untrusted interface in the network or to an interface on a device that is not in the network.

When a switch receives a packet on an untrusted interface and the interface belongs to a VLAN in which DHCP snooping is enabled, the switch compares the source MAC address and the DHCP client hardware address. If the addresses match (the default), the switch forwards the packet. If the addresses do not match, the switch drops the packet.

The switch drops a DHCP packet when one of these situations occurs:

- A packet from a DHCP server, such as a DHCP OFFER, DHCP ACK, DHCP NAK, or DHCP REQUEST packet, is received from outside the network or firewall.
- A packet is received on an untrusted interface, and the source MAC address and the DHCP client hardware address do not match.
- The switch receives a DHCP RELEASE or DHCP DECLINE broadcast message that has a MAC address in the DHCP snooping binding database, but the interface information in the binding database does not match the interface on which the message was received.
- A DHCP relay agent forwards a DHCP packet that includes a relay-agent IP address that is not 0.0.0.0, or the relay agent forwards a packet that includes option-82 information to an untrusted port.

If the switch is an aggregation switch supporting DHCP snooping and is connected to an edge switch that is inserting DHCP option-82 information, the switch drops packets with option-82 information when packets are received on an untrusted interface. If DHCP snooping is enabled and packets are received on a trusted port, the aggregation switch does not learn the DHCP snooping bindings for connected devices and cannot build a complete DHCP snooping binding database.

When an aggregation switch can be connected to an edge switch through an untrusted interface and you enter the **ip dhcp snooping information option allow-untrusted** global configuration command, the aggregation switch accepts packets with option-82 information from the edge switch. The aggregation switch learns the bindings for hosts connected through an untrusted switch interface. The DHCP security features, such as dynamic ARP inspection or IP source guard, can still be enabled on the aggregation switch while the switch receives packets with option-82 information on untrusted input interfaces to which hosts are connected. The port on the edge switch that connects to the aggregation switch must be configured as a trusted interface.

Normally, it is not desirable to broadcast packets to wireless clients. So, DHCP snooping replaces destination broadcast MAC address (ffff.ffff.ffff) with unicast MAC address for DHCP packets that are going from server to wireless clients. The unicast MAC address is retrieved from CHADDR field in the DHCP payload. This processing is applied for server to client packets such as DHCP OFFER, DHCP ACK, and DHCP NACK messages. The **ip dhcp snooping wireless bootp-broadcast enable** can be used to revert this behavior. When the wireless BOOTP broadcast is enabled, the broadcast DHCP packets from server are forwarded to wireless clients without changing the destination MAC address.

Option-82 Data Insertion

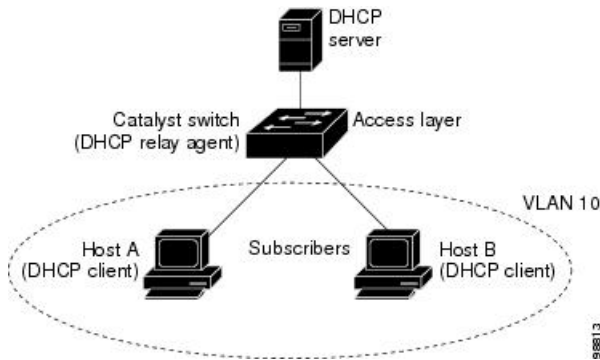
In residential, metropolitan Ethernet-access environments, DHCP can centrally manage the IP address assignments for a large number of subscribers. When the DHCP option-82 feature is enabled on the switch, a subscriber device is identified by the switch port through which it connects to the network (in addition to its MAC address). Multiple hosts on the subscriber LAN can be connected to the same port on the access switch and are uniquely identified.



Note The DHCP option-82 feature is supported only when DHCP snooping is globally enabled on the VLANs to which subscriber devices using option-82 are assigned.

The following illustration shows a metropolitan Ethernet network in which a centralized DHCP server assigns IP addresses to subscribers connected to the switch at the access layer. Because the DHCP clients and their associated DHCP server do not reside on the same IP network or subnet, a DHCP relay agent (the Catalyst switch) is configured with a helper address to enable broadcast forwarding and to transfer DHCP messages between the clients and the server.

Figure 104: DHCP Relay Agent in a Metropolitan Ethernet Network



When you enable the DHCP snooping information option 82 on the switch, the following sequence of events occurs:

- The host (DHCP client) generates a DHCP request and broadcasts it on the network.
- When the switch receives the DHCP request, it adds the option-82 information in the packet. By default, the remote-ID suboption is the switch MAC address, and the circuit-ID suboption is the port identifier, **vlan-mod-port**, from which the packet is received. You can configure the remote ID and circuit ID.
- If the IP address of the relay agent is configured, the switch adds this IP address in the DHCP packet.
- The switch forwards the DHCP request that includes the option-82 field to the DHCP server.
- The DHCP server receives the packet. If the server is option-82-capable, it can use the remote ID, the circuit ID, or both to assign IP addresses and implement policies, such as restricting the number of IP addresses that can be assigned to a single remote ID or circuit ID. Then the DHCP server echoes the option-82 field in the DHCP reply.
- The DHCP server unicasts the reply to the switch if the request was relayed to the server by the switch. The switch verifies that it originally inserted the option-82 data by inspecting the remote ID and possibly the circuit ID fields. The switch removes the option-82 field and forwards the packet to the switch port that connects to the DHCP client that sent the DHCP request.

In the default suboption configuration, when the described sequence of events occurs, the values in these fields do not change (see the illustration, *Suboption Packet Formats*):

- Circuit-ID suboption fields
 - Suboption type
 - Length of the suboption type
 - Circuit-ID type
 - Length of the circuit-ID type
- Remote-ID suboption fields
 - Suboption type
 - Length of the suboption type
 - Remote-ID type

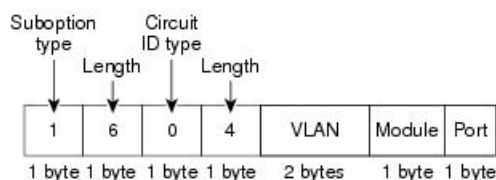
- Length of the remote-ID type

In the port field of the circuit ID suboption, the port numbers start at 3. For example, on a switch with 24 10/100/1000 ports and four small form-factor pluggable (SFP) module slots, port 3 is the Gigabit Ethernet 1/0/1 port, port 4 is the Gigabit Ethernet 1/0/2 port, and so forth. Port 27 is the SFP module slot Gigabit Ethernet 1/0/25, and so forth.

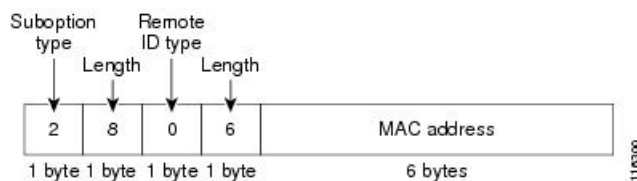
The illustration, *Suboption Packet Formats*, shows the packet formats for the remote-ID suboption and the circuit-ID suboption when the default suboption configuration is used. For the circuit-ID suboption, the module number corresponds to the switch number in the stack. The switch uses the packet formats when you globally enable DHCP snooping and enter the `ip dhcp snooping information option global` configuration command.

Figure 105: Suboption Packet Formats

Circuit ID Suboption Frame Format



Remote ID Suboption Frame Format

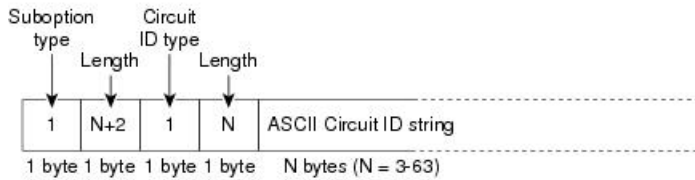
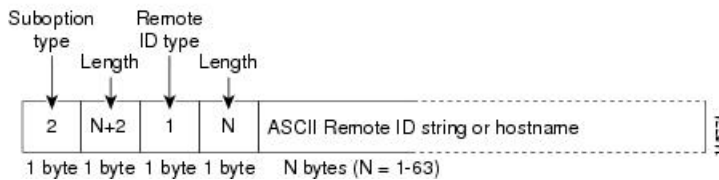


The illustration, *User-Configured Suboption Packet Formats*, shows the packet formats for user-configured remote-ID and circuit-ID suboptions. The switch uses these packet formats when DHCP snooping is globally enabled and when the `ip dhcp snooping information option format remote-id` global configuration command and the `ip dhcp snooping vlan information option format-type circuit-id string` interface configuration command are entered.

The values for these fields in the packets change from the default values when you configure the remote-ID and circuit-ID suboptions:

- Circuit-ID suboption fields
 - The circuit-ID type is 1.
 - The length values are variable, depending on the length of the string that you configure.
- Remote-ID suboption fields
 - The remote-ID type is 1.
 - The length values are variable, depending on the length of the string that you configure.

Figure 106: User-Configured Suboption Packet Formats

Circuit ID Suboption Frame Format (for user-configured string):**Remote ID Suboption Frame Format (for user-configured string):**

Cisco IOS DHCP Server Database

During the DHCP-based autoconfiguration process, the designated DHCP server uses the Cisco IOS DHCP server database. It has IP addresses, address bindings, and configuration parameters, such as the boot file.

An address binding is a mapping between an IP address and a MAC address of a host in the Cisco IOS DHCP server database. You can manually assign the client IP address, or the DHCP server can allocate an IP address from a DHCP address pool. For more information about manual and automatic address bindings, see the “Configuring DHCP” chapter of the *Cisco IOS IP Configuration Guide, Release 12.4*.

For procedures to enable and configure the Cisco IOS DHCP server database, see the “DHCP Configuration Task List” section in the “Configuring DHCP” chapter of the *Cisco IOS IP Configuration Guide, Release 12.4*.

DHCP Snooping Binding Database

When DHCP snooping is enabled, the switch uses the DHCP snooping binding database to store information about untrusted interfaces. The database can have up to 64,000 bindings.

Each database entry (binding) has an IP address, an associated MAC address, the lease time (in hexadecimal format), the interface to which the binding applies, and the VLAN to which the interface belongs. The database agent stores the bindings in a file at a configured location. At the end of each entry is a checksum that accounts for all the bytes from the start of the file through all the bytes associated with the entry. Each entry is 72 bytes, followed by a space and then the checksum value.

To keep the bindings when the switch reloads, you must use the DHCP snooping database agent. If the agent is disabled, dynamic ARP inspection or IP source guard is enabled, and the DHCP snooping binding database has dynamic bindings, the switch loses its connectivity. If the agent is disabled and only DHCP snooping is enabled, the switch does not lose its connectivity, but DHCP snooping might not prevent DHCP spoofing attacks.

When reloading, the switch reads the binding file to build the DHCP snooping binding database. The switch updates the file when the database changes.

When a switch learns of new bindings or when it loses bindings, the switch immediately updates the entries in the database. The switch also updates the entries in the binding file. The frequency at which the file is

updated is based on a configurable delay, and the updates are batched. If the file is not updated in a specified time (set by the write-delay and cancel-timeout values), the update stops.

This is the format of the file with bindings:

```
<initial-checksum>
TYPE DHCP-SNOOPING
VERSION 1
BEGIN
<entry-1> <checksum-1>
<entry-2> <checksum-1-2>
...
<entry-n> <checksum-1-2-..-n>
END
```

Each entry in the file is tagged with a checksum value that the switch uses to verify the entries when it reads the file. The initial-checksum entry on the first line distinguishes entries associated with the latest file update from entries associated with a previous file update.

This is an example of a binding file:

```
2bb4c2a1
TYPE DHCP-SNOOPING
VERSION 1
BEGIN
192.1.168.1 3 0003.47d8.c91f 2BB6488E Gi1/0/4 21ae5fbb
192.1.168.3 3 0003.44d6.c52f 2BB648EB Gi1/0/4 1bdb223f
192.1.168.2 3 0003.47d9.c8f1 2BB648AB Gi1/0/4 584a38f0
END
```

When the switch starts and the calculated checksum value equals the stored checksum value, the switch reads entries from the binding file and adds the bindings to its DHCP snooping binding database. The switch ignores an entry when one of these situations occurs:

- The switch reads the entry and the calculated checksum value does not equal the stored checksum value. The entry and the ones following it are ignored.
- An entry has an expired lease time (the switch might not remove a binding entry when the lease time expires).
- The interface in the entry no longer exists on the system.
- The interface is a routed interface or a DHCP snooping-trusted interface.

DHCP Snooping and Switch Stacks

DHCP snooping is managed on the active switch. When a new switch joins the stack, the switch receives the DHCP snooping configuration from the active switch. When a member switch leaves the stack, all DHCP snooping address bindings associated with the switch age out.

All snooping statistics are generated on the active switch. If a new active switch is elected, the statistics counters reset.

When a stack merge occurs, all DHCP snooping bindings in the active switch are lost if it is no longer the active switch. With a stack partition, the existing active switch is unchanged, and the bindings belonging to

the partitioned switches age out. The new active switch of the partitioned stack begins processing the new incoming DHCP packets.

How to Configure DHCP Features

Default DHCP Snooping Configuration

Table 138: Default DHCP Configuration

Feature	Default Setting
DHCP server	Enabled in Cisco IOS software, requires configuration ¹²
DHCP relay agent	Enabled ¹³
DHCP packet forwarding address	None configured
Checking the relay agent information	Enabled (invalid messages are dropped)
DHCP relay agent forwarding policy	Replace the existing relay agent information
DHCP snooping enabled globally	Disabled
DHCP snooping information option	Enabled
DHCP snooping option to accept packets on untrusted input interfaces ¹⁴	Disabled
DHCP snooping limit rate	None configured
DHCP snooping trust	Untrusted
DHCP snooping VLAN	Disabled
DHCP snooping MAC address verification	Enabled
Cisco IOS DHCP server binding database	Enabled in Cisco IOS software, requires configuration. Note The switch gets network addresses and configuration parameters only from a device configured as a DHCP server.
DHCP snooping binding database agent	Enabled in Cisco IOS software, requires configuration. This feature is operational only when a destination is configured.

¹² The switch responds to DHCP requests only if it is configured as a DHCP server.

¹³ The switch relays DHCP packets only if the IP address of the DHCP server is configured on the SVI of the DHCP client.

- ¹⁴ Use this feature when the switch is an aggregation switch that receives packets with option-82 information from an edge switch.

DHCP Snooping Configuration Guidelines

- If a switch port is connected to a DHCP server, configure a port as trusted by entering the **ip dhcp snooping trust interface** configuration command.
- If a switch port is connected to a DHCP client, configure a port as untrusted by entering the **no ip dhcp snooping trust** interface configuration command.
- You can display DHCP snooping statistics by entering the **show ip dhcp snooping statistics** user EXEC command, and you can clear the snooping statistics counters by entering the **clear ip dhcp snooping statistics** privileged EXEC command.

Configuring the DHCP Server

The switch can act as a DHCP server.

For procedures to configure the switch as a DHCP server, see the “Configuring DHCP” section of the “IP addressing and Services” section of the *Cisco IOS IP Configuration Guide, Release 12.4*.

DHCP Server and Switch Stacks

The DHCP binding database is managed on the stack's active switch. When a new active switch is assigned, the new active switch downloads the saved binding database from the TFTP server. When a switchover happens, the new active switch stack will use its database file that has been synced from the old active switch stack using the SSO function. The IP addresses associated with the lost bindings are released. You should configure an automatic backup by using the **ip dhcp database url [timeout seconds | write-delay seconds]** global configuration command.

Configuring the DHCP Relay Agent

Follow these steps to enable the DHCP relay agent on the switch:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **service dhcp**
4. **end**
5. **show running-config**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable	Enables privileged EXEC mode.

	Command or Action	Purpose
	Example: Device> enable	<ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	service dhcp Example: Device(config)# service dhcp	Enables the DHCP server and relay agent on your switch. By default, this feature is enabled.
Step 4	end Example: Device(config)# end	Returns to privileged EXEC mode.
Step 5	show running-config Example: Device# show running-config	Verifies your entries.
Step 6	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

What to do next

- Checking (validating) the relay agent information
- Configuring the relay agent forwarding policy

Specifying the Packet Forwarding Address

If the DHCP server and the DHCP clients are on different networks or subnets, you must configure the switch with the **ip helper-address** *address* interface configuration command. The general rule is to configure the command on the Layer 3 interface closest to the client. The address used in the **ip helper-address** command can be a specific DHCP server IP address, or it can be the network address if other DHCP servers are on the destination network segment. Using the network address enables any DHCP server to respond to requests.

Beginning in privileged EXEC mode, follow these steps to specify the packet forwarding address:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface vlan *vlan-id***
4. **ip address *ip-address subnet-mask***
5. **ip helper-address *address***
6. **end**
7. Use one of the following:
 - **interface range *port-range***
 - **interface *interface-id***
8. **switchport mode access**
9. **switchport access vlan *vlan-id***
10. **end**
11. **show running-config**
12. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	interface vlan <i>vlan-id</i> Example: Device(config)# interface vlan 1	Creates a switch virtual interface by entering a VLAN ID, and enter interface configuration mode.
Step 4	ip address <i>ip-address subnet-mask</i> Example: Device(config-if)# ip address 192.108.1.27 255.255.255.0	Configures the interface with an IP address and an IP subnet.
Step 5	ip helper-address <i>address</i> Example: Device(config-if)# ip helper-address 172.16.1.2	Specifies the DHCP packet forwarding address. The helper address can be a specific DHCP server address, or it can be the network address if other DHCP servers are on the destination network segment. Using the network address enables other servers to respond to DHCP requests.

	Command or Action	Purpose
		If you have multiple servers, you can configure one helper address for each server.
Step 6	end Example: Device(config-if)# end	Returns to global configuration mode.
Step 7	Use one of the following: <ul style="list-style-type: none"> • interface range <i>port-range</i> • interface <i>interface-id</i> Example: Device(config)# interface gigabitethernet1/0/2	Configures multiple physical ports that are connected to the DHCP clients, and enter interface range configuration mode. or Configures a single physical port that is connected to the DHCP client, and enter interface configuration mode.
Step 8	switchport mode access Example: Device(config-if)# switchport mode access	Defines the VLAN membership mode for the port.
Step 9	switchport access vlan <i>vlan-id</i> Example: Device(config-if)# switchport access vlan 1	Assigns the ports to the same VLAN as configured in Step 2.
Step 10	end Example: Device(config-if)# end	Returns to privileged EXEC mode.
Step 11	show running-config Example: Device# show running-config	Verifies your entries.
Step 12	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Prerequisites for Configuring DHCP Snooping and Option 82

The prerequisites for DHCP Snooping and Option 82 are as follows:

- You must globally enable DHCP snooping on the switch.

- Before globally enabling DHCP snooping on the switch, make sure that the devices acting as the DHCP server and the DHCP relay agent are configured and enabled.
- If you want the switch to respond to DHCP requests, it must be configured as a DHCP server.
- Before configuring the DHCP snooping information option on your switch, be sure to configure the device that is acting as the DHCP server. You must specify the IP addresses that the DHCP server can assign or exclude, or you must configure DHCP options for these devices.
- For DHCP snooping to function properly, all DHCP servers must be connected to the switch through trusted interfaces. In a service-provider network, a trusted interface is connected to a port on a device in the same network.
- You must configure the switch to use the Cisco IOS DHCP server binding database to use it for DHCP snooping.
- To use the DHCP snooping option of accepting packets on untrusted inputs, the switch must be an aggregation switch that receives packets with option-82 information from an edge switch.
- The following prerequisites apply to DHCP snooping binding database configuration:
 - You must configure a destination on the DHCP snooping binding database to use the switch for DHCP snooping.
 - Because both NVRAM and the flash memory have limited storage capacity, we recommend that you store the binding file on a TFTP server.
 - For network-based URLs (such as TFTP and FTP), you must create an empty file at the configured URL before the switch can write bindings to the binding file at that URL. See the documentation for your TFTP server to determine whether you must first create an empty file on the server; some TFTP servers cannot be configured this way.
 - To ensure that the lease time in the database is accurate, we recommend that you enable and configure Network Time Protocol (NTP).
 - If NTP is configured, the switch writes binding changes to the binding file only when the switch system clock is synchronized with NTP.
- Before configuring the DHCP relay agent on your switch, make sure to configure the device that is acting as the DHCP server. You must specify the IP addresses that the DHCP server can assign or exclude, configure DHCP options for devices, or set up the DHCP database agent.
- If you want the switch to relay DHCP packets, the IP address of the DHCP server must be configured on the switch virtual interface (SVI) of the DHCP client.
- If a switch port is connected to a DHCP server, configure a port as trusted by entering the **ip dhcp snooping trust interface** configuration command.
- If a switch port is connected to a DHCP client, configure a port as untrusted by entering the **no ip dhcp snooping trust** interface configuration command.

Enabling DHCP Snooping and Option 82

Follow these steps to enable DHCP snooping on the switch:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip dhcp snooping**
4. **ip dhcp snooping vlan *vlan-range***
5. **ip dhcp snooping information option**
6. **ip dhcp snooping information option format remote-id [string *ASCII-string* | hostname]**
7. **ip dhcp snooping information option allow-untrusted**
8. **interface *interface-id***
9. **ip dhcp snooping vlan *vlan* information option format-type circuit-id [override] string *ASCII-string***
10. **ip dhcp snooping trust**
11. **ip dhcp snooping limit rate *rate***
12. **exit**
13. **ip dhcp snooping verify mac-address**
14. **end**
15. **show running-config**
16. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	ip dhcp snooping Example: Device(config)# ip dhcp snooping	Enables DHCP snooping globally.
Step 4	ip dhcp snooping vlan <i>vlan-range</i> Example: Device(config)# ip dhcp snooping vlan 10	Enables DHCP snooping on a VLAN or range of VLANs. The range is 1 to 4094. You can enter a single VLAN ID identified by VLAN ID number, a series of VLAN IDs separated by commas, a range of VLAN IDs separated by hyphens, or a range of VLAN IDs separated by entering the starting and ending VLAN IDs separated by a space. <ul style="list-style-type: none"> • You can enter a single VLAN ID identified by VLAN ID number, a series of VLAN IDs separated by commas, a range of VLAN IDs separated by hyphens,

	Command or Action	Purpose
		or a range of VLAN IDs separated by entering the starting and ending VLAN IDs separated by a space.
Step 5	ip dhcp snooping information option Example: <pre>Device(config)# ip dhcp snooping information option</pre>	Enables the switch to insert and remove DHCP relay information (option-82 field) in forwarded DHCP request messages to the DHCP server. This is the default setting.
Step 6	ip dhcp snooping information option format remote-id [string ASCII-string hostname] Example: <pre>Device(config)# ip dhcp snooping information option format remote-id string acsiistring2</pre>	(Optional) Configures the remote-ID suboption. You can configure the remote ID as: <ul style="list-style-type: none"> • String of up to 63 ASCII characters (no spaces) • Configured hostname for the switch Note If the hostname is longer than 63 characters, it is truncated to 63 characters in the remote-ID configuration. The default remote ID is the switch MAC address.
Step 7	ip dhcp snooping information option allow-untrusted Example: <pre>Device(config)# ip dhcp snooping information option allow-untrusted</pre>	(Optional) If the switch is an aggregation switch connected to an edge switch, this command enables the switch to accept incoming DHCP snooping packets with option-82 information from the edge switch. The default setting is disabled. Note Enter this command only on aggregation switches that are connected to trusted devices.
Step 8	interface interface-id Example: <pre>Device(config)# interface gigabitethernet2/0/1</pre>	Specifies the interface to be configured, and enter interface configuration mode.
Step 9	ip dhcp snooping vlan vlan information option format-type circuit-id [override] string ASCII-string Example: <pre>Device(config-if)# ip dhcp snooping vlan 1 information option format-type circuit-id override string override2</pre>	(Optional) Configures the circuit-ID suboption for the specified interface. Specify the VLAN and port identifier, using a VLAN ID in the range of 1 to 4094. The default circuit ID is the port identifier, in the format vlan-mod-port . You can configure the circuit ID to be a string of 3 to 63 ASCII characters (no spaces). (Optional) Use the override keyword when you do not want the circuit-ID suboption inserted in TLV format to define subscriber information.
Step 10	ip dhcp snooping trust Example:	(Optional) Configures the interface as trusted or untrusted. Use the no keyword to configure an interface to receive

	Command or Action	Purpose
	Device(config-if)# <code>ip dhcp snooping trust</code>	messages from an untrusted client. The default setting is untrusted.
Step 11	ip dhcp snooping limit rate <i>rate</i> Example: Device(config-if)# <code>ip dhcp snooping limit rate 100</code>	(Optional) Configures the number of DHCP packets per second that an interface can receive. The range is 1 to 2048. By default, no rate limit is configured. Note We recommend an untrusted rate limit of not more than 100 packets per second. If you configure rate limiting for trusted interfaces, you might need to increase the rate limit if the port is a trunk port assigned to more than one VLAN with DHCP snooping.
Step 12	exit Example: Device(config-if)# <code>exit</code>	Returns to global configuration mode.
Step 13	ip dhcp snooping verify mac-address Example: Device(config)# <code>ip dhcp snooping verify mac-address</code>	(Optional) Configures the switch to verify that the source MAC address in a DHCP packet received on untrusted ports matches the client hardware address in the packet. The default is to verify that the source MAC address matches the client hardware address in the packet.
Step 14	end Example: Device(config)# <code>end</code>	Returns to privileged EXEC mode.
Step 15	show running-config Example: Device# <code>show running-config</code>	Verifies your entries.
Step 16	copy running-config startup-config Example: Device# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Enabling the Cisco IOS DHCP Server Database

For procedures to enable and configure the Cisco IOS DHCP server database, see the “DHCP Configuration Task List” section in the “Configuring DHCP” chapter of the Cisco IOS IP Configuration Guide, Release 12.4

Monitoring DHCP Snooping Information

Table 139: Commands for Displaying DHCP Information

show ip dhcp snooping	Displays the DHCP snooping configuration for a switch
show ip dhcp snooping binding	Displays only the dynamically configured bindings in the DHCP snooping binding database, also referred to as a binding table.
show ip dhcp snooping database	Displays the DHCP snooping binding database status and statistics.
show ip dhcp snooping statistics	Displays the DHCP snooping statistics in summary or detail form.
show ip source binding	Display the dynamically and statically configured bindings.



Note If DHCP snooping is enabled and an interface changes to the down state, the switch does not delete the statically configured bindings.

Configuring DHCP Server Port-Based Address Allocation

Information About Configuring DHCP Server Port-Based Address Allocation

DHCP server port-based address allocation is a feature that enables DHCP to maintain the same IP address on an Ethernet switch port regardless of the attached device client identifier or client hardware address.

When Ethernet switches are deployed in the network, they offer connectivity to the directly connected devices. In some environments, such as on a factory floor, if a device fails, the replacement device must be working immediately in the existing network. With the current DHCP implementation, there is no guarantee that DHCP would offer the same IP address to the replacement device. Control, monitoring, and other software expect a stable IP address associated with each device. If a device is replaced, the address assignment should remain stable even though the DHCP client has changed.

When configured, the DHCP server port-based address allocation feature ensures that the same IP address is always offered to the same connected port even as the client identifier or client hardware address changes in the DHCP messages received on that port. The DHCP protocol recognizes DHCP clients by the client identifier option in the DHCP packet. Clients that do not include the client identifier option are identified by the client hardware address. When you configure this feature, the port name of the interface overrides the client identifier or hardware address and the actual point of connection, the switch port, becomes the client identifier.

In all cases, by connecting the Ethernet cable to the same port, the same IP address is allocated through DHCP to the attached device.

The DHCP server port-based address allocation feature is only supported on a Cisco IOS DHCP server and not a third-party server.

Default Port-Based Address Allocation Configuration

By default, DHCP server port-based address allocation is disabled.

Port-Based Address Allocation Configuration Guidelines

- By default, DHCP server port-based address allocation is disabled.
- To restrict assignments from the DHCP pool to preconfigured reservations (unreserved addresses are not offered to the client and other clients are not served by the pool), you can enter the **reserved-only** DHCP pool configuration command.

Enabling the DHCP Snooping Binding Database Agent

Beginning in privileged EXEC mode, follow these steps to enable and configure the DHCP snooping binding database agent on the switch:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip dhcp snooping database** {flash[number]:/filename | ftp://user:password@host/filename | http://[[username:password]@]{hostname | host-ip}[/directory] /image-name.tar | rcp://user@host/filename} | **tftp://host/filename**
4. **ip dhcp snooping database timeout** seconds
5. **ip dhcp snooping database write-delay** seconds
6. **end**
7. **ip dhcp snooping binding** mac-address **vlan** vlan-id **ip-address** **interface** interface-id **expiry** seconds
8. **show ip dhcp snooping database** [detail]
9. **show running-config**
10. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example:	Enters global configuration mode.

	Command or Action	Purpose
	Device# <code>configure terminal</code>	
Step 3	<p>ip dhcp snooping database {flash[<i>number</i>]:/<i>filename</i> ftp://<i>user</i>:<i>password</i>@<i>host</i>/<i>filename</i> http://[[<i>username</i>:<i>password</i>]@]{<i>hostname</i> / <i>host-ip</i>}[/<i>directory</i>] <i>image-name.tar</i> rcp://<i>user</i>@<i>host</i>/<i>filename</i>} tftp://<i>host</i>/<i>filename</i></p> <p>Example:</p> <pre>Device(config)# ip dhcp snooping database tftp://10.90.90.90/snooping-rp2</pre>	<p>Specifies the URL for the database agent or the binding file by using one of these forms:</p> <ul style="list-style-type: none"> • flash[<i>number</i>]:/<i>filename</i> <p>(Optional) Use the <i>number</i> parameter to specify the stack member number of the active switch. The range for <i>number</i> is 1 to 9.</p> <ul style="list-style-type: none"> • ftp://<i>user</i>:<i>password</i>@<i>host</i>/<i>filename</i> • http://[[<i>username</i>:<i>password</i>]@]{<i>hostname</i> / <i>host-ip</i>}[/<i>directory</i>] <i>image-name.tar</i> • rcp://<i>user</i>@<i>host</i>/<i>filename</i> • tftp://<i>host</i>/<i>filename</i>
Step 4	<p>ip dhcp snooping database timeout <i>seconds</i></p> <p>Example:</p> <pre>Device(config)# ip dhcp snooping database timeout 300</pre>	<p>Specifies (in seconds) how long to wait for the database transfer process to finish before stopping the process.</p> <p>The default is 300 seconds. The range is 0 to 86400. Use 0 to define an infinite duration, which means to continue trying the transfer indefinitely.</p>
Step 5	<p>ip dhcp snooping database write-delay <i>seconds</i></p> <p>Example:</p> <pre>Device(config)# ip dhcp snooping database write-delay 15</pre>	<p>Specifies the duration for which the transfer should be delayed after the binding database changes. The range is from 15 to 86400 seconds. The default is 300 seconds (5 minutes).</p>
Step 6	<p>end</p> <p>Example:</p> <pre>Device(config)# end</pre>	<p>Returns to privileged EXEC mode.</p>
Step 7	<p>ip dhcp snooping binding <i>mac-address</i> vlan <i>vlan-id</i> <i>ip-address</i> interface <i>interface-id</i> expiry <i>seconds</i></p> <p>Example:</p> <pre>Device# ip dhcp snooping binding 0001.1234.1234 vlan 1 172.20.50.5 interface gil/1 expiry 1000</pre>	<p>(Optional) Adds binding entries to the DHCP snooping binding database. The <i>vlan-id</i> range is from 1 to 4904. The <i>seconds</i> range is from 1 to 4294967295.</p> <p>Enter this command for each entry that you add.</p> <p>Use this command when you are testing or debugging the switch.</p>
Step 8	<p>show ip dhcp snooping database [detail]</p> <p>Example:</p> <pre>Device# show ip dhcp snooping database detail</pre>	<p>Displays the status and statistics of the DHCP snooping binding database agent.</p>

	Command or Action	Purpose
Step 9	show running-config Example: Device# <code>show running-config</code>	Verifies your entries.
Step 10	copy running-config startup-config Example: Device# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Enabling DHCP Server Port-Based Address Allocation

Follow these steps to globally enable port-based address allocation and to automatically generate a subscriber identifier on an interface.

SUMMARY STEPS

1. `enable`
2. `configure terminal`
3. `ip dhcp use subscriber-id client-id`
4. `ip dhcp subscriber-id interface-name`
5. `interface interface-id`
6. `ip dhcp server use subscriber-id client-id`
7. `end`
8. `show running-config`
9. `copy running-config startup-config`

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> <code>enable</code>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode.

	Command or Action	Purpose
Step 3	ip dhcp use subscriber-id client-id Example: Device(config)# ip dhcp use subscriber-id client-id	Configures the DHCP server to globally use the subscriber identifier as the client identifier on all incoming DHCP messages.
Step 4	ip dhcp subscriber-id interface-name Example: Device(config)# ip dhcp subscriber-id interface-name	Automatically generates a subscriber identifier based on the short name of the interface. A subscriber identifier configured on a specific interface takes precedence over this command.
Step 5	interface interface-id Example: Device(config)# interface gigabitethernet1/0/1	Specifies the interface to be configured, and enter interface configuration mode.
Step 6	ip dhcp server use subscriber-id client-id Example: Device(config-if)# ip dhcp server use subscriber-id client-id	Configures the DHCP server to use the subscriber identifier as the client identifier on all incoming DHCP messages on the interface.
Step 7	end Example: Device(config)# end	Returns to privileged EXEC mode.
Step 8	show running-config Example: Device# show running-config	Verifies your entries.
Step 9	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

What to do next

After enabling DHCP port-based address allocation on the switch, use the **ip dhcp pool** global configuration command to preassign IP addresses and to associate them to clients.

Monitoring DHCP Server Port-Based Address Allocation

Table 140: Commands for Displaying DHCP Port-Based Address Allocation Information

Command	Purpose
show interface <i>interface id</i>	Displays the status and configuration of a specific interface.
show ip dhcp pool	Displays the DHCP address pools.
show ip dhcp binding	Displays address bindings on the Cisco IOS DHCP server.

Additional References

MIBs

MIB	MIBs Link
All the supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
<p>The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies.</p> <p>To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds.</p> <p>Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.</p>	http://www.cisco.com/support

Feature Information for DHCP Snooping and Option 82

Release	Feature Information
Cisco IOS Release 15.0(2)EX1	This feature was introduced.

Release	Feature Information
	<p>Introduced support for the following commands:</p> <ul style="list-style-type: none">• show ip dhcp snooping statistics user EXEC command for displaying DHCP snooping statistics.• clear ip dhcp snooping statistics privileged EXEC command for clearing the snooping statistics counters.



CHAPTER 72

Configuring IP Source Guard

IP Source Guard (IPSG) is a security feature that restricts IP traffic on nonrouted, Layer 2 interfaces by filtering traffic based on the DHCP snooping binding database and on manually configured IP source bindings.

This chapter contains the following topics:

- [Information About IP Source Guard, on page 1421](#)
- [How to Configure IP Source Guard, on page 1423](#)
- [Monitoring IP Source Guard, on page 1426](#)
- [Additional References, on page 1427](#)

Information About IP Source Guard

IP Source Guard

You can use IP source guard to prevent traffic attacks if a host tries to use the IP address of its neighbor and you can enable IP source guard when DHCP snooping is enabled on an untrusted interface.

After IPSG is enabled on an interface, the switch blocks all IP traffic received on the interface except for DHCP packets allowed by DHCP snooping.

The switch uses a source IP lookup table in hardware to bind IP addresses to ports. For IP and MAC filtering, a combination of source IP and source MAC lookups are used. IP traffic with a source IP address in the binding table is allowed, all other traffic is denied.

The IP source binding table has bindings that are learned by DHCP snooping or are manually configured (static IP source bindings). An entry in this table has an IP address, its associated MAC address, and its associated VLAN number. The switch uses the IP source binding table only when IP source guard is enabled.

IPSG is supported only on Layer 2 ports, including access and trunk ports. You can configure IPSG with source IP address filtering or with source IP and MAC address filtering.

IP Source Guard for Static Hosts



Note Do not use IPSG (IP source guard) for static hosts on uplink ports or trunk ports.

IPSG for static hosts extends the IPSG capability to non-DHCP and static environments. The previous IPSG used the entries created by DHCP snooping to validate the hosts connected to a switch. Any traffic received from a host without a valid DHCP binding entry is dropped. This security feature restricts IP traffic on nonrouted Layer 2 interfaces. It filters traffic based on the DHCP snooping binding database and on manually configured IP source bindings. The previous version of IPSG required a DHCP environment for IPSG to work.

IPSG for static hosts allows IPSG to work without DHCP. IPSG for static hosts relies on IP device tracking-table entries to install port ACLs. The switch creates static entries based on ARP requests or other IP packets to maintain the list of valid hosts for a given port. You can also specify the number of hosts allowed to send traffic to a given port. This is equivalent to port security at Layer 3.

IPSG for static hosts also supports dynamic hosts. If a dynamic host receives a DHCP-assigned IP address that is available in the IP DHCP snooping table, the same entry is learned by the IP device tracking table. In a stacked environment, when the active switch failover occurs, the IP source guard entries for static hosts attached to member ports are retained. When you enter the **show ip device tracking all EXEC** command, the IP device tracking table displays the entries as ACTIVE.



Note Some IP hosts with multiple network interfaces can inject some invalid packets into a network interface. The invalid packets contain the IP or MAC address for another network interface of the host as the source address. The invalid packets can cause IPSG for static hosts to connect to the host, to learn the invalid IP or MAC address bindings, and to reject the valid bindings. Consult the vendor of the corresponding operating system and the network interface to prevent the host from injecting invalid packets.

IPSG for static hosts initially learns IP or MAC bindings dynamically through an ACL-based snooping mechanism. IP or MAC bindings are learned from static hosts by ARP and IP packets. They are stored in the device tracking database. When the number of IP addresses that have been dynamically learned or statically configured on a given port reaches a maximum, the hardware drops any packet with a new IP address. To resolve hosts that have moved or gone away for any reason, IPSG for static hosts leverages IP device tracking to age out dynamically learned IP address bindings. This feature can be used with DHCP snooping. Multiple bindings are established on a port that is connected to both DHCP and static hosts. For example, bindings are stored in both the device tracking database as well as in the DHCP snooping binding database.

IP Source Guard Configuration Guidelines

- You can configure static IP bindings only on nonrouted ports. If you enter the **ip source binding mac-address vlan vlan-id ip-address interface interface-id** global configuration command on a routed interface, this error message appears:

```
Static IP source binding can only be configured on switch port.
```

- When IP source guard with source IP filtering is enabled on an interface, DHCP snooping must be enabled on the access VLAN for that interface.
- If you are enabling IP source guard on a trunk interface with multiple VLANs and DHCP snooping is enabled on all the VLANs, the source IP address filter is applied on all the VLANs.



Note If IP source guard is enabled and you enable or disable DHCP snooping on a VLAN on the trunk interface, the switch might not properly filter traffic.

- You can enable this feature when 802.1x port-based authentication is enabled.

How to Configure IP Source Guard

Enabling IP Source Guard

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **ip verify source** [**mac-check**]
5. **exit**
6. **ip source binding** *mac-address* **vlan** *vlan-id* *ip-address* **interface** *interface-id*
7. **end**
8. **show running-config**
9. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	interface <i>interface-id</i> Example: Device(config)# interface gigabitethernet 1/0/1	Specifies the interface to be configured, and enters interface configuration mode.
Step 4	ip verify source [mac-check]	Enables IP source guard with source IP address filtering.

	Command or Action	Purpose
	Example: Device(config-if)# ip verify source	(Optional) mac-check —Enables IP Source Guard with source IP address and MAC address filtering.
Step 5	exit Example: Device(config-if)# exit	Returns to global configuration mode.
Step 6	ip source binding mac-address vlan vlan-id ip-address interface interface-id Example: Device(config)# ip source binding 0100.0230.0002 vlan 11 10.0.0.4 interface gigabitethernet1/0/1	Adds a static IP source binding. Enter this command for each static binding.
Step 7	end Example: Device(config)# end	Returns to privileged EXEC mode.
Step 8	show running-config Example: Device# show running-config	Verifies your entries.
Step 9	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Configuring IP Source Guard for Static Hosts on a Layer 2 Access Port

You must configure the **ip device tracking maximum limit-number** interface configuration command globally for IPSG for static hosts to work. If you only configure this command on a port without enabling IP device tracking globally or by setting an IP device tracking maximum on that interface, IPSG with static hosts rejects all the IP traffic from that interface.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip device tracking**

4. **interface** *interface-id*
5. **switchport mode access**
6. **switchport access vlan** *vlan-id*
7. **ip verify source**[tracking] [mac-check]
8. **ip device tracking maximum** *number*
9. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	ip device tracking Example: Device(config)# ip device tracking	Turns on the IP host table, and globally enables IP device tracking.
Step 4	interface <i>interface-id</i> Example: Device(config)# interface gigabitethernet 1/0/1	Enters interface configuration mode.
Step 5	switchport mode access Example: Device(config-if)# switchport mode access	Configures a port as access.
Step 6	switchport access vlan <i>vlan-id</i> Example: Device(config-if)# switchport access vlan 10	Configures the VLAN for this port.
Step 7	ip verify source [tracking] [mac-check] Example:	Enables IP source guard with source IP address filtering. (Optional) tracking —Enables IP source guard for static hosts.

	Command or Action	Purpose
	Device(config-if)# ip verify source tracking mac-check	(Optional) mac-check —Enables MAC address filtering. The command ip verify source tracking mac-check enables IP source guard for static hosts with MAC address filtering.
Step 8	ip device tracking maximum <i>number</i> Example: Device(config-if)# ip device tracking maximum 8	Establishes a maximum limit for the number of static IPs that the IP device tracking table allows on the port. The range is 1 to 10. The maximum number is 10. Note You must configure the ip device tracking maximum <i>limit-number</i> interface configuration command.
Step 9	end Example: Device(config)# end	Returns to privileged EXEC mode.

Monitoring IP Source Guard

Table 141: Privileged EXEC show Commands

Command	Purpose
show ip verify source [interface <i>interface-id</i>]	Displays the IP source guard configuration on the switch or on a specific interface.
show ip device tracking { all interface <i>interface-id</i> ip <i>ip-address</i> mac <i>mac-address</i> }	Displays information about the entries in the IP device tracking table.

Table 142: Interface Configuration Commands

Command	Purpose
ip verify source tracking	Verifies the data source.

For detailed information about the fields in these displays, see the command reference for this release.

Additional References

Error Message Decoder

Description	Link
To help you research and resolve system error messages in this release, use the Error Message Decoder tool.	https://www.cisco.com/cgi-bin/Support/Errordecoder/index.cgi

MIBs

MIB	MIBs Link
All the supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
<p>The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies.</p> <p>To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds.</p> <p>Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.</p>	http://www.cisco.com/support



CHAPTER 73

Configuring Dynamic ARP Inspection

- [Restrictions for Dynamic ARP Inspection, on page 1429](#)
- [Understanding Dynamic ARP Inspection, on page 1430](#)
- [Default Dynamic ARP Inspection Configuration, on page 1434](#)
- [Relative Priority of ARP ACLs and DHCP Snooping Entries, on page 1434](#)
- [Configuring ARP ACLs for Non-DHCP Environments, on page 1434](#)
- [Configuring Dynamic ARP Inspection in DHCP Environments, on page 1437](#)
- [Limiting the Rate of Incoming ARP Packets, on page 1439](#)
- [Performing Dynamic ARP Inspection Validation Checks, on page 1442](#)
- [Monitoring DAI, on page 1443](#)
- [Verifying the DAI Configuration, on page 1444](#)
- [Additional References, on page 1444](#)

Restrictions for Dynamic ARP Inspection

This section lists the restrictions and guidelines for configuring Dynamic ARP Inspection on the switch.

- Dynamic ARP inspection is an ingress security feature; it does not perform any egress checking.
- Dynamic ARP inspection is not effective for hosts connected to switches that do not support dynamic ARP inspection or that do not have this feature enabled. Because man-in-the-middle attacks are limited to a single Layer 2 broadcast domain, separate the domain with dynamic ARP inspection checks from the one with no checking. This action secures the ARP caches of hosts in the domain enabled for dynamic ARP inspection.
- Dynamic ARP inspection depends on the entries in the DHCP snooping binding database to verify IP-to-MAC address bindings in incoming ARP requests and ARP responses. Make sure to enable DHCP snooping to permit ARP packets that have dynamically assigned IP addresses.

When DHCP snooping is disabled or in non-DHCP environments, use ARP ACLs to permit or to deny packets.

- Dynamic ARP inspection is supported on access ports, trunk ports, and EtherChannel ports.



Note Do not enable Dynamic ARP inspection on RSPAN VLANs. If Dynamic ARP inspection is enabled on RSPAN VLANs, Dynamic ARP inspection packets might not reach the RSPAN destination port.

- A physical port can join an EtherChannel port channel only when the trust state of the physical port and the channel port match. Otherwise, the physical port remains suspended in the port channel. A port channel inherits its trust state from the first physical port that joins the channel. Consequently, the trust state of the first physical port need not match the trust state of the channel.

Conversely, when you change the trust state on the port channel, the switch configures a new trust state on all the physical ports that comprise the channel.

- The rate limit is calculated separately on each switch in a switch stack. For a cross-stack EtherChannel, this means that the actual rate limit might be higher than the configured value. For example, if you set the rate limit to 30 pps on an EtherChannel that has one port on switch 1 and one port on switch 2, each port can receive packets at 29 pps without causing the EtherChannel to become error-disabled.
- The operating rate for the port channel is cumulative across all the physical ports within the channel. For example, if you configure the port channel with an ARP rate-limit of 400 pps, all the interfaces combined on the channel receive an aggregate 400 pps. The rate of incoming ARP packets on EtherChannel ports is equal to the sum of the incoming rate of packets from all the channel members. Configure the rate limit for EtherChannel ports only after examining the rate of incoming ARP packets on the channel-port members.

The rate of incoming packets on a physical port is checked against the port-channel configuration rather than the physical-ports configuration. The rate-limit configuration on a port channel is independent of the configuration on its physical ports.

If the EtherChannel receives more ARP packets than the configured rate, the channel (including all physical ports) is placed in the error-disabled state.

- Make sure to limit the rate of ARP packets on incoming trunk ports. Configure trunk ports with higher rates to reflect their aggregation and to handle packets across multiple dynamic ARP inspection-enabled VLANs. You also can use the **ip arp inspection limit none** interface configuration command to make the rate unlimited. A high rate-limit on one VLAN can cause a denial-of-service attack to other VLANs when the software places the port in the error-disabled state.
- When you enable dynamic ARP inspection on the switch, policers that were configured to police ARP traffic are no longer effective. The result is that all ARP traffic is sent to the CPU.
- In the presence of vlan-bridging & IP device tracking, the cross-stack ARP packet forwarding will not work.

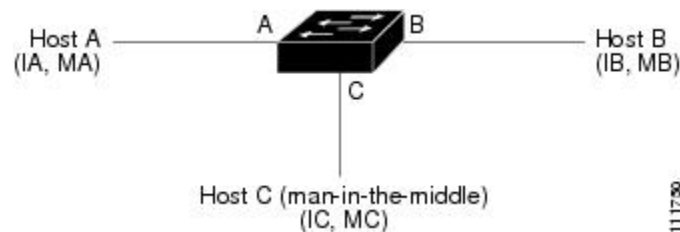
Understanding Dynamic ARP Inspection

ARP provides IP communication within a Layer 2 broadcast domain by mapping an IP address to a MAC address. For example, Host B wants to send information to Host A but does not have the MAC address of Host A in its ARP cache. Host B generates a broadcast message for all hosts within the broadcast domain to obtain the MAC address associated with the IP address of Host A. All hosts within the broadcast domain receive the ARP request, and Host A responds with its MAC address. However, because ARP allows a gratuitous

reply from a host even if an ARP request was not received, an ARP spoofing attack and the poisoning of ARP caches can occur. After the attack, all traffic from the device under attack flows through the attacker's computer and then to the router, switch, or host.

A malicious user can attack hosts, switches, and routers connected to your Layer 2 network by poisoning the ARP caches of systems connected to the subnet and by intercepting traffic intended for other hosts on the subnet. Figure 26-1 shows an example of ARP cache poisoning.

Figure 107: ARP Cache Poisoning



Hosts A, B, and C are connected to the switch on interfaces A, B and C, all of which are on the same subnet. Their IP and MAC addresses are shown in parentheses; for example, Host A uses IP address IA and MAC address MA. When Host A needs to communicate to Host B at the IP layer, it broadcasts an ARP request for the MAC address associated with IP address IB. When the switch and Host B receive the ARP request, they populate their ARP caches with an ARP binding for a host with the IP address IA and a MAC address MA; for example, IP address IA is bound to MAC address MA. When Host B responds, the switch and Host A populate their ARP caches with a binding for a host with the IP address IB and the MAC address MB.

Host C can poison the ARP caches of the switch, Host A, and Host B by broadcasting forged ARP responses with bindings for a host with an IP address of IA (or IB) and a MAC address of MC. Hosts with poisoned ARP caches use the MAC address MC as the destination MAC address for traffic intended for IA or IB. This means that Host C intercepts that traffic. Because Host C knows the true MAC addresses associated with IA and IB, it can forward the intercepted traffic to those hosts by using the correct MAC address as the destination. Host C has inserted itself into the traffic stream from Host A to Host B, the classic *man-in-the-middle* attack.

Dynamic ARP inspection is a security feature that validates ARP packets in a network. It intercepts, logs, and discards ARP packets with invalid IP-to-MAC address bindings. This capability protects the network from certain man-in-the-middle attacks.

Dynamic ARP inspection ensures that only valid ARP requests and responses are relayed. The switch performs these activities:

- Intercepts all ARP requests and responses on untrusted ports
- Verifies that each of these intercepted packets has a valid IP-to-MAC address binding before updating the local ARP cache or before forwarding the packet to the appropriate destination
- Drops invalid ARP packets

Dynamic ARP inspection determines the validity of an ARP packet based on valid IP-to-MAC address bindings stored in a trusted database, the DHCP snooping binding database. This database is built by DHCP snooping if DHCP snooping is enabled on the VLANs and on the switch. If the ARP packet is received on a trusted interface, the switch forwards the packet without any checks. On untrusted interfaces, the switch forwards the packet only if it is valid.

You enable dynamic ARP inspection on a per-VLAN basis by using the **ip arp inspection vlan *vlan-range*** global configuration command.

In non-DHCP environments, dynamic ARP inspection can validate ARP packets against user-configured ARP access control lists (ACLs) for hosts with statically configured IP addresses. You define an ARP ACL by using the **arp access-list** *acl-name* global configuration command.

You can configure dynamic ARP inspection to drop ARP packets when the IP addresses in the packets are invalid or when the MAC addresses in the body of the ARP packets do not match the addresses specified in the Ethernet header. Use the **ip arp inspection validate** {[src-mac] [dst-mac] [ip]} global configuration command.

Interface Trust States and Network Security

Dynamic ARP inspection associates a trust state with each interface on the switch. Packets arriving on trusted interfaces bypass all dynamic ARP inspection validation checks, and those arriving on untrusted interfaces undergo the dynamic ARP inspection validation process.

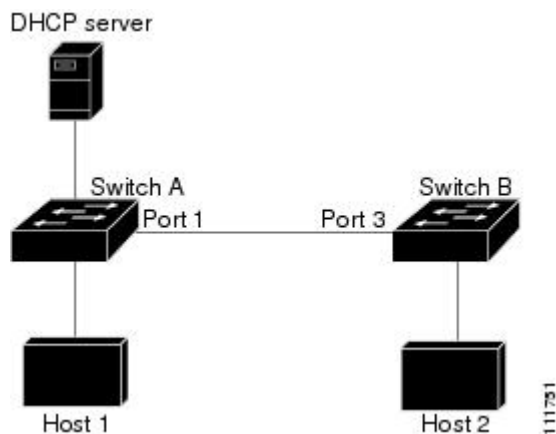
In a typical network configuration, you configure all switch ports connected to host ports as untrusted and configure all switch ports connected to switches as trusted. With this configuration, all ARP packets entering the network from a given switch bypass the security check. No other validation is needed at any other place in the VLAN or in the network. You configure the trust setting by using the `arp inspection trust interface` configuration command.



Caution Use the trust state configuration carefully. Configuring interfaces as untrusted when they should be trusted can result in a loss of connectivity.

In the following figure, assume that both Switch A and Switch B are running dynamic ARP inspection on the VLAN that includes Host 1 and Host 2. If Host 1 and Host 2 acquire their IP addresses from the DHCP server connected to Switch A, only Switch A binds the IP-to-MAC address of Host 1. Therefore, if the interface between Switch A and Switch B is untrusted, the ARP packets from Host 1 are dropped by Switch B. Connectivity between Host 1 and Host 2 is lost.

Figure 108: ARP Packet Validation on a VLAN Enabled for Dynamic ARP Inspection



Configuring interfaces to be trusted when they are actually untrusted leaves a security hole in the network. If Switch A is not running dynamic ARP inspection, Host 1 can easily poison the ARP cache of Switch B (and Host 2, if the link between the switches is configured as trusted). This condition can occur even though Switch B is running dynamic ARP inspection.

Dynamic ARP inspection ensures that hosts (on untrusted interfaces) connected to a switch running dynamic ARP inspection do not poison the ARP caches of other hosts in the network. However, dynamic ARP inspection does not prevent hosts in other portions of the network from poisoning the caches of the hosts that are connected to a switch running dynamic ARP inspection.

In cases in which some switches in a VLAN run dynamic ARP inspection and other switches do not, configure the interfaces connecting such switches as untrusted. However, to validate the bindings of packets from nondynamic ARP inspection switches, configure the switch running dynamic ARP inspection with ARP ACLs. When you cannot determine such bindings, at Layer 3, isolate switches running dynamic ARP inspection from switches not running dynamic ARP inspection switches.



Note Depending on the setup of the DHCP server and the network, it might not be possible to validate a given ARP packet on all switches in the VLAN.

Rate Limiting of ARP Packets

The switch CPU performs dynamic ARP inspection validation checks; therefore, the number of incoming ARP packets is rate-limited to prevent a denial-of-service attack. By default, the rate for untrusted interfaces is 15 packets per second (pps). Trusted interfaces are not rate-limited. You can change this setting by using the **ip arp inspection limit** interface configuration command.

When the rate of incoming ARP packets exceeds the configured limit, the switch places the port in the error-disabled state. The port remains in that state until you intervene. You can use the **errdisable recovery** global configuration command to enable error disable recovery so that ports automatically emerge from this state after a specified timeout period.



Note The rate limit for an EtherChannel is applied separately to each switch in a stack. For example, if a limit of 20 pps is configured on the EtherChannel, each switch with ports in the EtherChannel can carry up to 20 pps. If any switch exceeds the limit, the entire EtherChannel is placed into the error-disabled state.

Relative Priority of ARP ACLs and DHCP Snooping Entries

Dynamic ARP inspection uses the DHCP snooping binding database for the list of valid IP-to-MAC address bindings.

ARP ACLs take precedence over entries in the DHCP snooping binding database. The switch uses ACLs only if you configure them by using the **ip arp inspection filter vlan** global configuration command. The switch first compares ARP packets to user-configured ARP ACLs. If the ARP ACL denies the ARP packet, the switch also denies the packet even if a valid binding exists in the database populated by DHCP snooping.

Logging of Dropped Packets

When the switch drops a packet, it places an entry in the log buffer and then generates system messages on a rate-controlled basis. After the message is generated, the switch clears the entry from the log buffer. Each log entry contains flow information, such as the receiving VLAN, the port number, the source and destination IP addresses, and the source and destination MAC addresses.

You use the **ip arp inspection log-buffer** global configuration command to configure the number of entries in the buffer and the number of entries needed in the specified interval to generate system messages. You specify the type of packets that are logged by using the **ip arp inspection vlan logging** global configuration command.

Default Dynamic ARP Inspection Configuration

Feature	Default Settings
Dynamic ARP inspection	Disabled on all VLANs.
Interface trust state	All interfaces are untrusted.
Rate limit of incoming ARP packets	The rate is 15 pps on untrusted interfaces, assuming that the network is a switched network with a host connecting to as many as 15 new hosts per second. The rate is unlimited on all trusted interfaces. The burst interval is 1 second.
ARP ACLs for non-DHCP environments	No ARP ACLs are defined.
Validation checks	No checks are performed.
Log buffer	When dynamic ARP inspection is enabled, all denied or dropped ARP packets are logged. The number of entries in the log is 32. The number of system messages is limited to 5 per second. The logging-rate interval is 1 second.
Per-VLAN logging	All denied or dropped ARP packets are logged.

Relative Priority of ARP ACLs and DHCP Snooping Entries

Dynamic ARP inspection uses the DHCP snooping binding database for the list of valid IP-to-MAC address bindings.

ARP ACLs take precedence over entries in the DHCP snooping binding database. The switch uses ACLs only if you configure them by using the **ip arp inspection filter vlan** global configuration command. The switch first compares ARP packets to user-configured ARP ACLs. If the ARP ACL denies the ARP packet, the switch also denies the packet even if a valid binding exists in the database populated by DHCP snooping.

Configuring ARP ACLs for Non-DHCP Environments

This procedure shows how to configure dynamic ARP inspection when Switch B shown in Figure 2 does not support dynamic ARP inspection or DHCP snooping.

If you configure port 1 on Switch A as trusted, a security hole is created because both Switch A and Host 1 could be attacked by either Switch B or Host 2. To prevent this possibility, you must configure port 1 on Switch A as untrusted. To permit ARP packets from Host 2, you must set up an ARP ACL and apply it to VLAN 1. If the IP address of Host 2 is not static (it is impossible to apply the ACL configuration on Switch A) you must separate Switch A from Switch B at Layer 3 and use a router to route packets between them.

Follow these steps to configure an ARP ACL on Switch A. This procedure is required in non-DHCP environments.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **arp access-list *acl-name***
4. **permit ip host *sender-ip* mac host *sender-mac***
5. **exit**
6. **ip arp inspection filter *arp-acl-name* vlan *vlan-range* [static]**
7. **interface *interface-id***
8. **no ip arp inspection trust**
9. **end**
10. Use the following show commands:
 - **show arp access-list *acl-name***
 - **show ip arp inspection vlan *vlan-range***
 - **show ip arp inspection interfaces**
11. **show running-config**
12. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	arp access-list <i>acl-name</i>	Defines an ARP ACL, and enters ARP access-list configuration mode. By default, no ARP access lists are defined. Note At the end of the ARP access list, there is an implicit deny ip any mac any command.

	Command or Action	Purpose
Step 4	permit ip host <i>sender-ip</i> mac host <i>sender-mac</i>	Permits ARP packets from the specified host (Host 2). <ul style="list-style-type: none"> • For <i>sender-ip</i>, enter the IP address of Host 2. • For <i>sender-mac</i>, enter the MAC address of Host 2.
Step 5	exit	Returns to global configuration mode.
Step 6	ip arp inspection filter <i>arp-acl-name</i> vlan <i>vlan-range</i> [static]	Applies ARP ACL to the VLAN. By default, no defined ARP ACLs are applied to any VLAN. <ul style="list-style-type: none"> • For <i>arp-acl-name</i>, specify the name of the ACL created in Step 2. • For <i>vlan-range</i>, specify the VLAN that the switches and hosts are in. You can specify a single VLAN identified by VLAN ID number, a range of VLANs separated by a hyphen, or a series of VLANs separated by a comma. The range is 1 to 4094. • (Optional) Specify static to treat implicit denies in the ARP ACL as explicit denies and to drop packets that do not match any previous clauses in the ACL. DHCP bindings are not used. <p>If you do not specify this keyword, it means that there is no explicit deny in the ACL that denies the packet, and DHCP bindings determine whether a packet is permitted or denied if the packet does not match any clauses in the ACL.</p> <p>ARP packets containing only IP-to-MAC address bindings are compared against the ACL. Packets are permitted only if the access list permits them.</p>
Step 7	interface <i>interface-id</i>	Specifies Switch A interface that is connected to Switch B, and enters the interface configuration mode.
Step 8	no ip arp inspection trust	Configures Switch A interface that is connected to Switch B as untrusted. <p>By default, all interfaces are untrusted.</p> <p>For untrusted interfaces, the switch intercepts all ARP requests and responses. It verifies that the intercepted packets have valid IP-to-MAC address bindings before updating the local cache and before forwarding the packet to the appropriate destination. The switch drops invalid packets and logs them in the log buffer according to the logging configuration specified with the ip arp inspection vlan logging global configuration command.</p>
Step 9	end	Returns to privileged EXEC mode.

	Command or Action	Purpose
Step 10	Use the following show commands: <ul style="list-style-type: none"> • show arp access-list <i>acl-name</i> • show ip arp inspection vlan <i>vlan-range</i> • show ip arp inspection interfaces 	Verifies your entries.
Step 11	show running-config Example: Device# show running-config	Verifies your entries.
Step 12	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Configuring Dynamic ARP Inspection in DHCP Environments

Before you begin

This procedure shows how to configure dynamic ARP inspection when two switches support this feature. Host 1 is connected to Switch A, and Host 2 is connected to Switch B. Both switches are running dynamic ARP inspection on VLAN 1 where the hosts are located. A DHCP server is connected to Switch A. Both hosts acquire their IP addresses from the same DHCP server. Therefore, Switch A has the bindings for Host 1 and Host 2, and Switch B has the binding for Host 2.



Note Dynamic ARP inspection depends on the entries in the DHCP snooping binding database to verify IP-to-MAC address bindings in incoming ARP requests and ARP responses. Make sure to enable DHCP snooping to permit ARP packets that have dynamically assigned IP addresses.

Follow these steps to configure dynamic ARP inspection. You must perform this procedure on both switches. This procedure is required.

SUMMARY STEPS

1. **enable**
2. **show cdp neighbors**
3. **configure terminal**
4. **ip arp inspection vlan** *vlan-range*
5. **Interface***interface-id*
6. **ip arp inspection trust**
7. **end**
8. **show ip arp inspection interfaces**

9. **show ip arp inspection vlan** *vlan-range*
10. **show ip dhcp snooping binding**
11. **show ip arp inspection statistics vlan** *vlan-range*
12. **configure terminal**
13. **configure terminal**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	show cdp neighbors Example: Device (config-if) # show cdp neighbors	Verify the connection between the switches.
Step 3	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 4	ip arp inspection vlan <i>vlan-range</i> Example: Device (config) # ip arp inspection vlan 1	Enable dynamic ARP inspection on a per-VLAN basis. By default, dynamic ARP inspection is disabled on all VLANs. For <i>vlan-range</i> , specify a single VLAN identified by VLAN ID number, a range of VLANs separated by a hyphen, or a series of VLANs separated by a comma. The range is 1 to 4094. Specify the same VLAN ID for both switches.
Step 5	Interface <i>interface-id</i> Example: Device (config) # interface gigabitethernet1/0/1	Specifies the interface connected to the other switch, and enter interface configuration mode.
Step 6	ip arp inspection trust Example: Device (config-if) # ip arp inspection trust	Configures the connection between the switches as trusted. By default, all interfaces are untrusted. The switch does not check ARP packets that it receives from the other switch on the trusted interface. It simply forwards the packets. For untrusted interfaces, the switch intercepts all ARP requests and responses. It verifies that the intercepted packets have valid IP-to-MAC address bindings before updating the local cache and before forwarding the packet to the appropriate destination. The switch drops invalid packets and logs them in the log buffer according to the

	Command or Action	Purpose
		logging configuration specified with the ip arp inspection vlan logging global configuration command.
Step 7	end Example: Device(config-if)#end	Returns to privileged EXEC mode.
Step 8	show ip arp inspection interfaces Example:	Verifies the dynamic ARP inspection configuration on interfaces.
Step 9	show ip arp inspection vlan <i>vlan-range</i> Example: Device(config-if)#show ip arp inspection vlan 1	Verifies the dynamic ARP inspection configuration on VLAN.
Step 10	show ip dhcp snooping binding Example: Device(config-if)#show ip dhcp snooping binding	Verifies the DHCP bindings.
Step 11	show ip arp inspection statistics vlan <i>vlan-range</i> Example: Device(config-if)#show ip arp inspection statistics vlan 1	Checks the dynamic ARP inspection statistics on VLAN.
Step 12	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 13	configure terminal Example: Device# configure terminal	Enters global configuration mode.

Limiting the Rate of Incoming ARP Packets

The switch CPU performs dynamic ARP inspection validation checks; therefore, the number of incoming ARP packets is rate-limited to prevent a denial-of-service attack.

When the rate of incoming ARP packets exceeds the configured limit, the switch places the port in the error-disabled state. The port remains in that state until you enable error-disabled recovery so that ports automatically emerge from this state after a specified timeout period.



Note Unless you configure a rate limit on an interface, changing the trust state of the interface also changes its rate limit to the default value for that trust state. After you configure the rate limit, the interface retains the rate limit even when its trust state is changed. If you enter the **no ip arp inspection limit** interface configuration command, the interface reverts to its default rate limit.

Follow these steps to limit the rate of incoming ARP packets. This procedure is optional.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **ip arp inspection limit** {rate pps [burst interval seconds] | none}
5. **exit**
6. Use the following commands:
 - **errdisable detect cause arp-inspection**
 - **errdisable recovery cause arp-inspection**
 - **errdisable recovery interval** *interval*
7. **exit**
8. Use the following show commands:
 - **show ip arp inspection interfaces**
 - **show errdisable recovery**
9. **show running-config**
10. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	interface <i>interface-id</i>	Specifies the interface to be rate-limited, and enter interface configuration mode.
Step 4	ip arp inspection limit {rate pps [burst interval seconds] none}	Limits the rate of incoming ARP requests and responses on the interface. The default rate is 15 pps on untrusted

	Command or Action	Purpose
		<p>interfaces and unlimited on trusted interfaces. The burst interval is 1 second.</p> <p>The keywords have these meanings:</p> <ul style="list-style-type: none"> • For rate<i>pps</i>, specify an upper limit for the number of incoming packets processed per second. The range is 0 to 2048 pps. • (Optional) For burst interval<i>seconds</i>, specify the consecutive interval in seconds, over which the interface is monitored for a high rate of ARP packets. The range is 1 to 15. • For rate none, specify no upper limit for the rate of incoming ARP packets that can be processed.
Step 5	exit	Returns to global configuration mode.
Step 6	<p>Use the following commands:</p> <ul style="list-style-type: none"> • errdisable detect cause arp-inspection • errdisable recovery cause arp-inspection • errdisable recovery interval <i>interval</i> 	<p>(Optional) Enables error recovery from the dynamic ARP inspection error-disabled state, and configure the dynamic ARP inspection recover mechanism variables.</p> <p>By default, recovery is disabled, and the recovery interval is 300 seconds.</p> <p>For interval <i>interval</i>, specify the time in seconds to recover from the error-disabled state. The range is 30 to 86400.</p>
Step 7	exit	Returns to privileged EXEC mode.
Step 8	<p>Use the following show commands:</p> <ul style="list-style-type: none"> • show ip arp inspection interfaces • show errdisable recovery 	Verifies your settings.
Step 9	<p>show running-config</p> <p>Example:</p> <pre>Device# show running-config</pre>	Verifies your entries.
Step 10	<p>copy running-config startup-config</p> <p>Example:</p> <pre>Device# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Performing Dynamic ARP Inspection Validation Checks

Dynamic ARP inspection intercepts, logs, and discards ARP packets with invalid IP-to-MAC address bindings. You can configure the switch to perform additional checks on the destination MAC address, the sender and target IP addresses, and the source MAC address.

Follow these steps to perform specific checks on incoming ARP packets. This procedure is optional.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip arp inspection validate {[src-mac] [dst-mac] [ip]}**
4. **exit**
5. **show ip arp inspection vlan *vlan-range***
6. **show running-config**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	ip arp inspection validate {[src-mac] [dst-mac] [ip]}	Performs a specific check on incoming ARP packets. By default, no checks are performed. The keywords have these meanings: <ul style="list-style-type: none"> • For src-mac, check the source MAC address in the Ethernet header against the sender MAC address in the ARP body. This check is performed on both ARP requests and responses. When enabled, packets with different MAC addresses are classified as invalid and are dropped. • For dst-mac, check the destination MAC address in the Ethernet header against the target MAC address in ARP body. This check is performed for ARP responses. When enabled, packets with different MAC addresses are classified as invalid and are dropped.

	Command or Action	Purpose
		<ul style="list-style-type: none"> For ip, check the ARP body for invalid and unexpected IP addresses. Addresses include 0.0.0.0, 255.255.255.255, and all IP multicast addresses. Sender IP addresses are checked in all ARP requests and responses, and target IP addresses are checked only in ARP responses. <p>You must specify at least one of the keywords. Each command overrides the configuration of the previous command; that is, if a command enables src and dst mac validations, and a second command enables IP validation only, the src and dst mac validations are disabled as a result of the second command.</p>
Step 4	exit	Returns to privileged EXEC mode.
Step 5	show ip arp inspection vlan <i>vlan-range</i>	Verifies your settings.
Step 6	show running-config Example: Device# <code>show running-config</code>	Verifies your entries.
Step 7	copy running-config startup-config Example: Device# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Monitoring DAI

To monitor DAI, use the following commands:

Command	Description
clear ip arp inspection statistics	Clears dynamic ARP inspection statistics.
show ip arp inspection statistics [vlan <i>vlan-range</i>]	Displays statistics for forwarded, dropped, MAC validation failure, IP validation failure, ACL permitted and denied, and DHCP permitted and denied packets for the specified VLAN. If no VLANs are specified or if a range is specified, displays information only for VLANs with dynamic ARP inspection enabled (active).
clear ip arp inspection log	Clears the dynamic ARP inspection log buffer.
show ip arp inspection log	Displays the configuration and contents of the dynamic ARP inspection log buffer.

For the **show ip arp inspection statistics** command, the switch increments the number of forwarded packets for each ARP request and response packet on a trusted dynamic ARP inspection port. The switch increments the number of ACL or DHCP permitted packets for each packet that is denied by source MAC, destination MAC, or IP validation checks, and the switch increments the appropriate.

Verifying the DAI Configuration

To display and verify the DAI configuration, use the following commands:

Command	Description
show arp access-list [<i>acl-name</i>]	Displays detailed information about ARP ACLs.
show ip arp inspection interfaces [<i>interface-id</i>]	Displays the trust state and the rate limit of ARP packets for the specified interface or all interfaces.
show ip arp inspection vlan <i>vlan-range</i>	Displays the configuration and the operating state of dynamic ARP inspection for the specified VLAN. If no VLANs are specified or if a range is specified, displays information only for VLANs with dynamic ARP inspection enabled (active).

Additional References

Error Message Decoder

Description	Link
To help you research and resolve system error messages in this release, use the Error Message Decoder tool.	https://www.cisco.com/cgi-bin/Support/Errordecoder/index.cgi

MIBs

MIB	MIBs Link
All the supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
<p>The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies.</p> <p>To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds.</p> <p>Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.</p>	<p>http://www.cisco.com/support</p>



CHAPTER 74

Configuring IEEE 802.1x Port-Based Authentication

This chapter describes how to configure IEEE 802.1x port-based authentication. IEEE 802.1x authentication prevents unauthorized devices (clients) from gaining access to the network. Unless otherwise noted, the term *switch* refers to a standalone switch or a switch stack.

- [Information About 802.1x Port-Based Authentication, on page 1447](#)
- [How to Configure 802.1x Port-Based Authentication, on page 1480](#)
- [Monitoring 802.1x Statistics and Status, on page 1536](#)
- [Additional References for IEEE 802.1x Port-Based Authentication, on page 1536](#)
- [Feature Information for 802.1x Port-Based Authentication, on page 1537](#)

Information About 802.1x Port-Based Authentication

The 802.1x standard defines a client-server-based access control and authentication protocol that prevents unauthorized clients from connecting to a LAN through publicly accessible ports unless they are properly authenticated. The authentication server authenticates each client connected to a switch port before making available any services offered by the switch or the LAN.



Note TACACS is not supported with 802.1x authentication.

Until the client is authenticated, 802.1x access control allows only Extensible Authentication Protocol over LAN (EAPOL), Cisco Discovery Protocol (CDP), and Spanning Tree Protocol (STP) traffic through the port to which the client is connected. After authentication is successful, normal traffic can pass through the port.

Client session	Maximum sessions supported
Maximum dot1x or MAB client sessions	2000
Maximum web-based authentication sessions	2000
Maximum dot1x sessions with critical-auth VLAN enabled and server re-initialized	2000

Client session	Maximum sessions supported
Maximum MAB sessions with various session features applied	2000
Maximum dot1x sessions with service templates or session features applied	2000

Port-Based Authentication Process

To configure IEEE 802.1X port-based authentication, you must enable authentication, authorization, and accounting (AAA) and specify the authentication method list. A method list describes the sequence and authentication method to be queried to authenticate a user.

The AAA process begins with authentication. When 802.1x port-based authentication is enabled and the client supports 802.1x-compliant client software, these events occur:

- If the client identity is valid and the 802.1x authentication succeeds, the switch grants the client access to the network.
- If 802.1x authentication times out while waiting for an EAPOL message exchange and MAC authentication bypass is enabled, the switch can use the client MAC address for authorization. If the client MAC address is valid and the authorization succeeds, the switch grants the client access to the network. If the client MAC address is invalid and the authorization fails, the switch assigns the client to a guest VLAN that provides limited services if a guest VLAN is configured.
- If the switch gets an invalid identity from an 802.1x-capable client and a restricted VLAN is specified, the switch can assign the client to a restricted VLAN that provides limited services.
- If the RADIUS authentication server is unavailable (down) and inaccessible authentication bypass is enabled, the switch grants the client access to the network by putting the port in the critical-authentication state in the RADIUS-configured or the user-specified access VLAN.

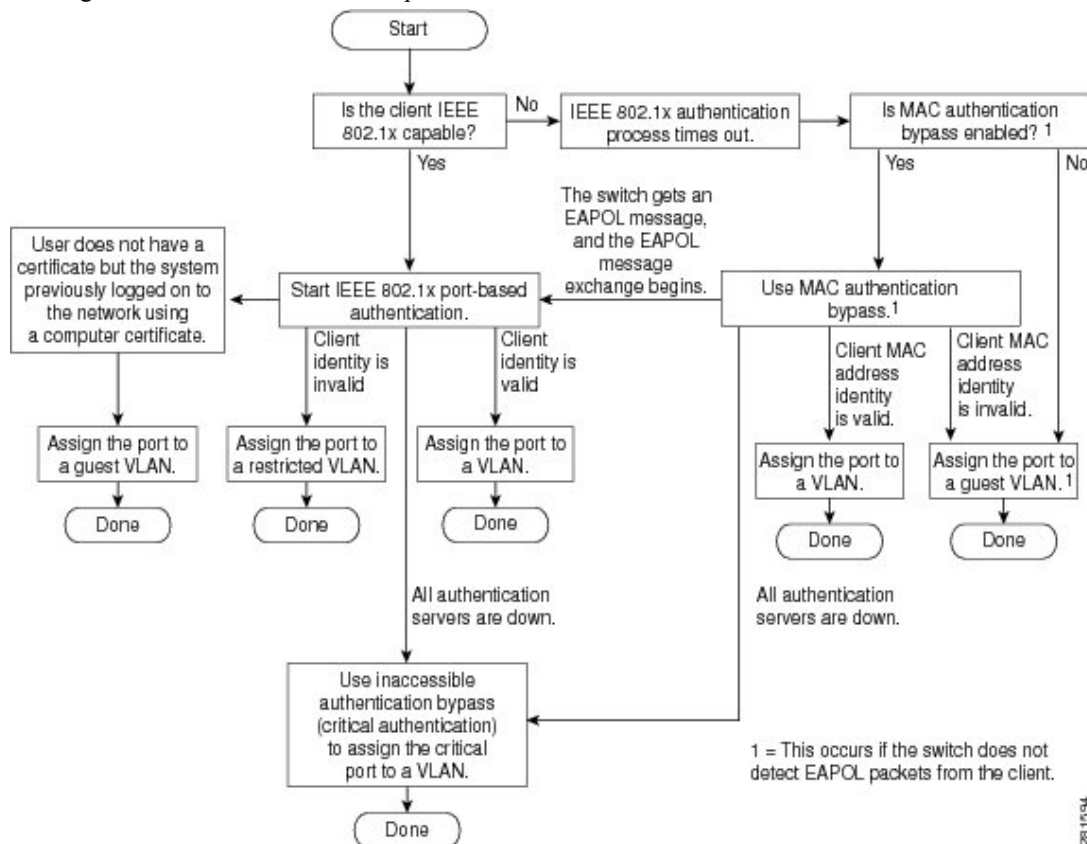


Note Inaccessible authentication bypass is also referred to as critical authentication or the AAA fail policy.

If Multi Domain Authentication (MDA) is enabled on a port, this flow can be used with some exceptions that are applicable to voice authorization.

Figure 109: Authentication Flowchart

This figure shows the authentication process.



The switch re-authenticates a client when one of these situations occurs:

- Periodic re-authentication is enabled, and the re-authentication timer expires.

You can configure the re-authentication timer to use a switch-specific value or to be based on values from the RADIUS server.

After 802.1x authentication using a RADIUS server is configured, the switch uses timers based on the Session-Timeout RADIUS attribute (Attribute[27]) and the Termination-Action RADIUS attribute (Attribute [29]).

The Session-Timeout RADIUS attribute (Attribute[27]) specifies the time after which re-authentication occurs.

The Termination-Action RADIUS attribute (Attribute [29]) specifies the action to take during re-authentication. The actions are *Initialize* and *ReAuthenticate*. When the *Initialize* action is set (the attribute value is *DEFAULT*), the 802.1x session ends, and connectivity is lost during re-authentication. When the *ReAuthenticate* action is set (the attribute value is *RADIUS-Request*), the session is not affected during re-authentication.

- You manually re-authenticate the client by entering the **dot1x re-authenticate interface interface-id** privileged EXEC command.

Port-Based Authentication Initiation and Message Exchange

During 802.1x authentication, the switch or the client can initiate authentication. If you enable authentication on a port by using the **authentication port-control auto** interface configuration command, the switch initiates authentication when the link state changes from down to up or periodically as long as the port remains up and unauthenticated. The switch sends an EAP-request/identity frame to the client to request its identity. Upon receipt of the frame, the client responds with an EAP-response/identity frame.

However, if during bootup, the client does not receive an EAP-request/identity frame from the switch, the client can initiate authentication by sending an EAPOL-start frame, which prompts the switch to request the client's identity.



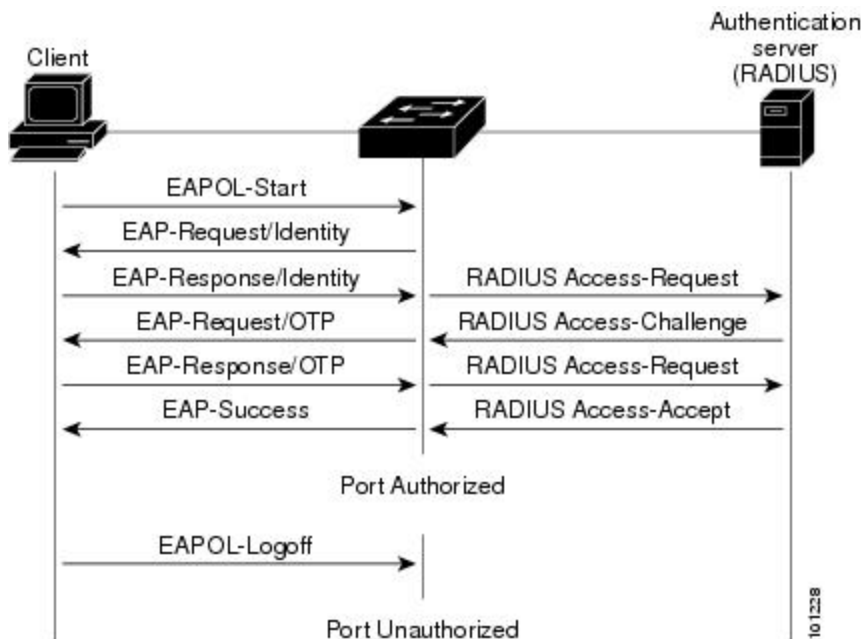
Note If 802.1x authentication is not enabled or supported on the network access device, any EAPOL frames from the client are dropped. If the client does not receive an EAP-request/identity frame after three attempts to start authentication, the client sends frames as if the port is in the authorized state. A port in the authorized state effectively means that the client has been successfully authenticated.

When the client supplies its identity, the switch begins its role as the intermediary, passing EAP frames between the client and the authentication server until authentication succeeds or fails. If the authentication succeeds, the switch port becomes authorized. If the authentication fails, authentication can be retried, the port might be assigned to a VLAN that provides limited services, or network access is not granted.

The specific exchange of EAP frames depends on the authentication method being used.

Figure 110: Message Exchange

This figure shows a message exchange initiated by the client when the client uses the One-Time-Password (OTP) authentication method with a RADIUS server.

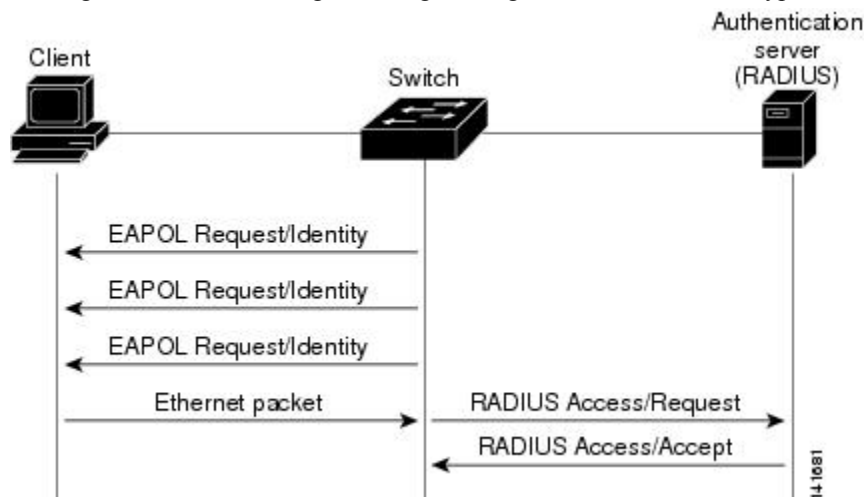


If 802.1x authentication times out while waiting for an EAPOL message exchange and MAC authentication bypass is enabled, the switch can authorize the client when the switch detects an Ethernet packet from the

client. The switch uses the MAC address of the client as its identity and includes this information in the RADIUS-access/request frame that is sent to the RADIUS server. After the server sends the switch the RADIUS-access/accept frame (authorization is successful), the port becomes authorized. If authorization fails and a guest VLAN is specified, the switch assigns the port to the guest VLAN. If the switch detects an EAPOL packet while waiting for an Ethernet packet, the switch stops the MAC authentication bypass process and starts 802.1x authentication.

Figure 111: Message Exchange During MAC Authentication Bypass

This figure shows the message exchange during MAC authentication bypass.



Authentication Manager for Port-Based Authentication

Port-Based Authentication Methods

Table 143: 802.1x Features

Authentication method	Mode			
	Single host	Multiple host	MDA	Multiple Authentication
802.1x	VLAN assignment Per-user ACL Filter-ID attribute Downloadable ACL Redirect URL	VLAN assignment	VLAN assignment Per-user ACL Filter-ID attribute Downloadable ACL Redirect URL	VLAN assignment Per-user ACL Filter-ID attribute Downloadable ACL Redirect URL

Authentication method	Mode			
	Single host	Multiple host	MDA	Multiple Authentication
MAC authentication bypass	VLAN assignment Per-user ACL Filter-ID attribute Downloadable ACL Redirect URL	VLAN assignment	VLAN assignment Per-user ACL Filter-Id attribute Downloadable ACL Redirect URL	VLAN assignment Per-user ACL Filter-Id attribute Downloadable ACL Redirect URL
Standalone web authentication	Proxy ACL, Filter-Id attribute, downloadable ACL			
NAC Layer 2 IP validation	Filter-Id attribute Downloadable ACL Redirect URL	Filter-Id attribute Downloadable ACL Redirect URL	Filter-Id attribute Downloadable ACL Redirect URL	Filter-Id attribute Downloadable ACL Redirect URL
Web authentication as fallback method	Proxy ACL Filter-Id attribute Downloadable ACL	Proxy ACL Filter-Id attribute Downloadable ACL	Proxy ACL Filter-Id attribute Downloadable ACL	Proxy ACL Filter-Id attribute Downloadable ACL

¹⁵ Supported in Cisco IOS Release 12.2(50)SE and later.

¹⁶ For clients that do not support 802.1x authentication.

Per-User ACLs and Filter-Ids



Note You can only set **any** as the source in the ACL.



Note For any ACL configured for multiple-host mode, the source portion of statement must be *any*. (For example, **permit icmp any host 10.10.1.1**.)

You must specify **any** in the source ports of any defined ACL. Otherwise, the ACL cannot be applied and authorization fails. Single host is the only exception to support backward compatibility.

More than one host can be authenticated on MDA-enabled and multiauth ports. The ACL policy applied for one host does not effect the traffic of another host. If only one host is authenticated on a multi-host port, and the other hosts gain network access without authentication, the ACL policy for the first host can be applied to the other connected hosts by specifying any in the source address.

Port-Based Authentication Manager CLI Commands

The authentication-manager interface-configuration commands control all the authentication methods, such as 802.1x, MAC authentication bypass, and web authentication. The authentication manager commands determine the priority and order of authentication methods applied to a connected host.

The authentication manager commands control generic authentication features, such as host-mode, violation mode, and the authentication timer. Generic authentication commands include the **authentication host-mode**, **authentication violation**, and **authentication timer** interface configuration commands.

802.1x-specific commands begin with the **dot1x** keyword. For example, the **authentication port-control auto** interface configuration command enables authentication on an interface.

To disable dot1x on a switch, remove the configuration globally by using the `no dot1x system-auth-control`, and also remove it from all configured interfaces.



Note If 802.1x authentication is globally disabled, other authentication methods are still enabled on that port, such as web authentication.

The **authentication manager** commands provide the same functionality as earlier 802.1x commands.

When filtering out verbose system messages generated by the authentication manager, the filtered content typically relates to authentication success. You can also filter verbose messages for 802.1x authentication and MAB authentication. There is a separate command for each authentication method:

- The **no authentication logging verbose** global configuration command filters verbose messages from the authentication manager.
- The **no dot1x logging verbose** global configuration command filters 802.1x authentication verbose messages.
- The **no mab logging verbose** global configuration command filters MAC authentication bypass (MAB) verbose messages

Table 144: Authentication Manager Commands and Earlier 802.1x Commands

The authentication manager commands in Cisco IOS Release 12.2(50)SE or later	The equivalent 802.1x commands in Cisco IOS Release 12.2(46)SE and earlier	Description
authentication control-direction {both in}	dot1x control-direction {both in}	Enable 802.1x authentication with the wake-on-LAN (WoL) feature, and configure the port control as unidirectional or bidirectional.
authentication event	dot1x auth-fail vlan dot1x critical (interface configuration) dot1x guest-vlan6	Enable the restricted VLAN on a port. Enable the inaccessible-authentication-bypass feature. Specify an active VLAN as an 802.1x guest VLAN.

The authentication manager commands in Cisco IOS Release 12.2(50)SE or later	The equivalent 802.1x commands in Cisco IOS Release 12.2(46)SE and earlier	Description
authentication fallback <i>fallback-profile</i>	dot1x fallback <i>fallback-profile</i>	Configure a port to use web authentication as a fallback method for clients that do not support 802.1x authentication.
authentication host-mode [multi-auth multi-domain multi-host single-host]	dot1x host-mode { single-host multi-host multi-domain }	Allow a single host (client) or multiple hosts on an 802.1x-authorized port.
authentication order	mab	Provides the flexibility to define the order of authentication methods to be used.
authentication periodic	dot1x reauthentication	Enable periodic re-authentication of the client.
authentication port-control { auto force-authorized force-unauthorized }	dot1x port-control { auto force-authorized force-unauthorized }	Enable manual control of the authorization state of the port.
authentication timer	dot1x timeout	Set the 802.1x timers.
authentication violation { protect restrict shutdown }	dot1x violation-mode { shutdown restrict protect }	Configure the violation modes that occur when a new device connects to a port or when a new device connects to a port after the maximum number of devices are connected to that port.

Ports in Authorized and Unauthorized States

During 802.1x authentication, depending on the switch port state, the switch can grant a client access to the network. The port starts in the *unauthorized* state. While in this state, the port that is not configured as a voice VLAN port disallows all ingress and egress traffic except for 802.1x authentication, CDP, and STP packets. When a client is successfully authenticated, the port changes to the *authorized* state, allowing all traffic for the client to flow normally. If the port is configured as a voice VLAN port, the port allows VoIP traffic and 802.1x protocol packets before the client is successfully authenticated.



Note CDP bypass is not supported and may cause a port to go into err-disabled state.

If a client that does not support 802.1x authentication connects to an unauthorized 802.1x port, the switch requests the client's identity. In this situation, the client does not respond to the request, the port remains in the unauthorized state, and the client is not granted access to the network.

In contrast, when an 802.1x-enabled client connects to a port that is not running the 802.1x standard, the client initiates the authentication process by sending the EAPOL-start frame. When no response is received, the

client sends the request for a fixed number of times. Because no response is received, the client begins sending frames as if the port is in the authorized state.

You control the port authorization state by using the **authentication port-control** interface configuration command and these keywords:

- **force-authorized**—disables 802.1x authentication and causes the port to change to the authorized state without any authentication exchange required. The port sends and receives normal traffic without 802.1x-based authentication of the client. This is the default setting.
- **force-unauthorized**—causes the port to remain in the unauthorized state, ignoring all attempts by the client to authenticate. The switch cannot provide authentication services to the client through the port.
- **auto**—enables 802.1x authentication and causes the port to begin in the unauthorized state, allowing only EAPOL frames to be sent and received through the port. The authentication process begins when the link state of the port changes from down to up or when an EAPOL-start frame is received. The switch requests the identity of the client and begins relaying authentication messages between the client and the authentication server. Each client attempting to access the network is uniquely identified by the switch by using the client MAC address.

If the client is successfully authenticated (receives an Accept frame from the authentication server), the port state changes to authorized, and all frames from the authenticated client are allowed through the port. If the authentication fails, the port remains in the unauthorized state, but authentication can be retried. If the authentication server cannot be reached, the switch can resend the request. If no response is received from the server after the specified number of attempts, authentication fails, and network access is not granted.

When a client logs off, it sends an EAPOL-logoff message, causing the switch port to change to the unauthorized state.

If the link state of a port changes from up to down, or if an EAPOL-logoff frame is received, the port returns to the unauthorized state.

Port-Based Authentication and Switch Stacks

If a switch is added to or removed from a switch stack, 802.1x authentication is not affected as long as the IP connectivity between the RADIUS server and the stack remains intact. This statement also applies if the stack's active switch is removed from the switch stack. Note that if the active switch fails, a stack member becomes the new active switch of the stack by using the election process, and the 802.1x authentication process continues as usual.

If IP connectivity to the RADIUS server is interrupted because the switch that was connected to the server is removed or fails, these events occur:

- Ports that are already authenticated and that do not have periodic re-authentication enabled remain in the authenticated state. Communication with the RADIUS server is not required.
- Ports that are already authenticated and that have periodic re-authentication enabled (with the **dot1x re-authentication** global configuration command) fail the authentication process when the re-authentication occurs. Ports return to the unauthenticated state during the re-authentication process. Communication with the RADIUS server is required.

For an ongoing authentication, the authentication fails immediately because there is no server connectivity.

If the switch that failed comes up and rejoins the switch stack, the authentications might or might not fail depending on the boot-up time and whether the connectivity to the RADIUS server is re-established by the time the authentication is attempted.

To avoid loss of connectivity to the RADIUS server, you should ensure that there is a redundant connection to it. For example, you can have a redundant connection to the stack's active switch and another to a stack member, and if the active switch fails, the switch stack still has connectivity to the RADIUS server.

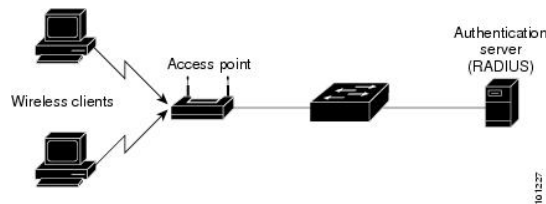
802.1x Host Mode

You can configure an 802.1x port for single-host or for multiple-hosts mode. In single-host mode, only one client can be connected to the 802.1x-enabled switch port. The switch detects the client by sending an EAPOL frame when the port link state changes to the up state. If a client leaves or is replaced with another client, the switch changes the port link state to down, and the port returns to the unauthorized state.

In multiple-hosts mode, you can attach multiple hosts to a single 802.1x-enabled port. In this mode, only one of the attached clients must be authorized for all clients to be granted network access. If the port becomes unauthorized (re-authentication fails or an EAPOL-logoff message is received), the switch denies network access to all of the attached clients.

In this topology, the wireless access point is responsible for authenticating the clients attached to it, and it also acts as a client to the switch.

Figure 112: Multiple Host Mode Example



Note For all host modes, the line protocol stays up before authorization when port-based authentication is configured.

The switch supports multidomain authentication (MDA), which allows both a data device and a voice device, such as an IP Phone (Cisco or non-Cisco), to connect to the same switch port.

802.1x Multiple Authentication Mode

Multiple-authentication (multiauth) mode allows multiple authenticated clients on the data VLAN and voice VLAN. Each host is individually authenticated. There is no limit to the number of data or voice device that can be authenticated on a multiauthport.

If a hub or access point is connected to an 802.1x-enabled port, each connected client must be authenticated. For non-802.1x devices, you can use MAC authentication bypass or web authentication as the per-host authentication fallback method to authenticate different hosts with different methods on a single port.



Note When a port is in multiple-authentication mode, the authentication-failed VLAN features do not activate.

You can assign a RADIUS-server-supplied VLAN in multi-auth mode, under the following conditions:

- The host is the first host authorized on the port, and the RADIUS server supplies VLAN information
- Subsequent hosts are authorized with a VLAN that matches the operational VLAN.
- A host is authorized on the port with no VLAN assignment, and subsequent hosts either have no VLAN assignment, or their VLAN information matches the operational VLAN.
- The first host authorized on the port has a group VLAN assignment, and subsequent hosts either have no VLAN assignment, or their group VLAN matches the group VLAN on the port. Subsequent hosts must use the same VLAN from the VLAN group as the first host. If a VLAN list is used, all hosts are subject to the conditions specified in the VLAN list.
- After a VLAN is assigned to a host on the port, subsequent hosts must have matching VLAN information or be denied access to the port.
- You cannot configure a guest VLAN or an auth-fail VLAN in multi-auth mode.
- The behavior of the critical-auth VLAN is not changed for multi-auth mode. When a host tries to authenticate and the server is not reachable, all authorized hosts are reinitialized in the configured VLAN.

Multi-auth Per User VLAN assignment

The Multi-auth Per User VLAN assignment feature allows you to create multiple operational access VLANs based on VLANs assigned to the clients on the port that has a single configured access VLAN. The port configured as an access port where the traffic for all the VLANs associated with data domain is not dot1q tagged, and these VLANs are treated as native VLANs.

The number of hosts per multi-auth port is 8, however there can be more hosts.

The following scenarios are associated with the multi-auth Per User VLAN assignments:

Scenario one

When a hub is connected to an access port, and the port is configured with an access VLAN (V0).

The host (H1) is assigned to VLAN (V1) through the hub. The operational VLAN of the port is changed to V1. This behaviour is similar on a single-host or multi-domain-auth port.

When a second host (H2) is connected and gets assigned to VLAN (V2), the port will have two operational VLANs (V1 and V2). If H1 and H2 sends untagged ingress traffic, H1 traffic is mapped to VLAN (V1) and H2 traffic to VLAN (V2), all egress traffic going out of the port on VLAN (V1) and VLAN (V2) are untagged.

If both the hosts, H1 and H2 are logged out or the sessions are removed due to some reason then VLAN (V1) and VLAN (V2) are removed from the port, and the configured VLAN (V0) is restored on the port.

Scenario two

When a hub is connected to an access port, and the port is configured with an access VLAN (V0). The host (H1) is assigned to VLAN (V1) through the hub. The operational VLAN of the port is changed to V1.

When a second host (H2) is connected and gets authorized without explicit vlan policy, H2 is expected to use the configured VLAN (V0) that is restored on the port. All egress traffic going out of two operational VLANs, VLAN (V0) and VLAN (V1) are untagged.

If host (H2) is logged out or the session is removed due to some reason then the configured VLAN (V0) is removed from the port, and VLAN (V1) becomes the only operational VLAN on the port.

Scenario three

When a hub is connected to an access port in open mode, and the port is configured with an access VLAN (V0).

The host (H1) is assigned to VLAN (V1) through the hub. The operational VLAN of the port is changed to V1. When a second host (H2) is connected and remains unauthorized, it still has access to operational VLAN (V1) due to open mode.

If host H1 is logged out or the session is removed due to some reason, VLAN (V1) is removed from the port and host (H2) gets assigned to VLAN (V0).



Note The combination of Open mode and VLAN assignment has an adverse affect on host (H2) because it has an IP address in the subnet that corresponds to VLAN (V1).

Limitation in Multi-auth Per User VLAN assignment

In the Multi-auth Per User VLAN assignment feature, egress traffic from multiple vlans are untagged on a port where the hosts receive traffic that is not meant for them. This can be a problem with broadcast and multicast traffic.

- **IPv4 ARPs:** Hosts receive ARP packets from other subnets. This is a problem if two subnets in different Virtual Routing and Forwarding (VRF) tables with overlapping IP address range are active on the port. The host ARP cache may get invalid entries.
- **IPv6 control packets:** In IPv6 deployments, Router Advertisements (RA) are processed by hosts that are not supposed to receive them. When a host from one VLAN receives RA from a different VLAN, the host assign incorrect IPv6 address to itself. Such a host is unable to get access to the network.

The workaround is to enable the IPv6 first hop security so that the broadcast ICMPv6 packets are converted to unicast and sent out from multi-auth enabled ports.. The packet is replicated for each client in multi-auth port belonging to the VLAN and the destination MAC is set to an individual client. Ports having one VLAN, ICMPv6 packets broadcast normally.

- **IP multicast:** Multicast traffic destined to a multicast group gets replicated for different VLANs if the hosts on those VLANs join the multicast group. When two hosts in different VLANs join a multicast group (on the same mutli-auth port), two copies of each multicast packet are sent out from that port.

MAC Move

When a MAC address is authenticated on one switch port, that address is not allowed on another authentication manager-enabled port of the switch. If the switch detects that same MAC address on another authentication manager-enabled port, the address is not allowed.

There are situations where a MAC address might need to move from one port to another on the same switch. For example, when there is another device (for example a hub or an IP phone) between an authenticated host and a switch port, you might want to disconnect the host from the device and connect it directly to another port on the same switch.

You can globally enable MAC move so the device is reauthenticated on the new port. When a host moves to a second port, the session on the first port is deleted, and the host is reauthenticated on the new port. MAC move is supported on all host modes. (The authenticated host can move to any port on the switch, no matter which host mode is enabled on the that port.) When a MAC address moves from one port to another, the

switch terminates the authenticated session on the original port and initiates a new authentication sequence on the new port. The MAC move feature applies to both voice and data hosts.



Note In open authentication mode, a MAC address is immediately moved from the original port to the new port, with no requirement for authorization on the new port.

MAC Replace

The MAC replace feature can be configured to address the violation that occurs when a host attempts to connect to a port where another host was previously authenticated.



Note This feature does not apply to ports in multi-auth mode, because violations are not triggered in that mode. It does not apply to ports in multiple host mode, because in that mode, only the first host requires authentication.

If you configure the **authentication violation** interface configuration command with the **replace** keyword, the authentication process on a port in multi-domain mode is:

- A new MAC address is received on a port with an existing authenticated MAC address.
- The authentication manager replaces the MAC address of the current data host on the port with the new MAC address.
- The authentication manager initiates the authentication process for the new MAC address.
- If the authentication manager determines that the new host is a voice host, the original voice host is removed.

If a port is in open authentication mode, any new MAC address is immediately added to the MAC address table.

802.1x Accounting

The 802.1x standard defines how users are authorized and authenticated for network access but does not keep track of network usage. 802.1x accounting is disabled by default. You can enable 802.1x accounting to monitor this activity on 802.1x-enabled ports:

- User successfully authenticates.
- User logs off.
- Link-down occurs.
- Re-authentication successfully occurs.
- Re-authentication fails.

The switch does not log 802.1x accounting information. Instead, it sends this information to the RADIUS server, which must be configured to log accounting messages.

802.1x Accounting Attribute-Value Pairs

The information sent to the RADIUS server is represented in the form of Attribute-Value (AV) pairs. These AV pairs provide data for different applications. (For example, a billing application might require information that is in the Acct-Input-Octets or the Acct-Output-Octets attributes of a RADIUS packet.)

AV pairs are automatically sent by a switch that is configured for 802.1x accounting. Three types of RADIUS accounting packets are sent by a switch:

- START—sent when a new user session starts
- INTERIM—sent during an existing session for updates
- STOP—sent when a session terminates

You can view the AV pairs that are being sent by the switch by entering the **debug radius accounting** privileged EXEC command. For more information about this command, see the *Cisco IOS Debug Command Reference, Release 12.4*.

This table lists the AV pairs and when they are sent are sent by the switch.

Table 145: Accounting AV Pairs

Attribute Number	AV Pair Name	START	INTERIM	STOP
Attribute[1]	User-Name	Always	Always	Always
Attribute[4]	NAS-IP-Address	Always	Always	Always
Attribute[5]	NAS-Port	Always	Always	Always
Attribute[8]	Framed-IP-Address	Never	Sometimes ¹⁷	Sometimes
Attribute[25]	Class	Always	Always	Always
Attribute[30]	Called-Station-ID	Always	Always	Always
Attribute[31]	Calling-Station-ID	Always	Always	Always
Attribute[40]	Acct-Status-Type	Always	Always	Always
Attribute[41]	Acct-Delay-Time	Always	Always	Always
Attribute[42]	Acct-Input-Octets	Never	Always	Always
Attribute[43]	Acct-Output-Octets	Never	Always	Always
Attribute[47]	Acct-Input-Packets	Never	Always	Always
Attribute[48]	Acct-Output-Packets	Never	Always	Always
Attribute[44]	Acct-Session-ID	Always	Always	Always
Attribute[45]	Acct-Authentic	Always	Always	Always
Attribute[46]	Acct-Session-Time	Never	Always	Always

Attribute Number	AV Pair Name	START	INTERIM	STOP
Attribute[49]	Acct-Terminate-Cause	Never	Never	Always
Attribute[61]	NAS-Port-Type	Always	Always	Always

¹⁷ The Framed-IP-Address AV pair is sent when a valid static IP address is configured or when a Dynamic Host Control Protocol (DHCP) binding exists for the host in the DHCP snooping bindings table.

802.1x Readiness Check

The 802.1x readiness check monitors 802.1x activity on all the switch ports and displays information about the devices connected to the ports that support 802.1x. You can use this feature to determine if the devices connected to the switch ports are 802.1x-capable. You use an alternate authentication such as MAC authentication bypass or web authentication for the devices that do not support 802.1x functionality.

This feature only works if the supplicant on the client supports a query with the NOTIFY EAP notification packet. The client must respond within the 802.1x timeout value.

Switch-to-RADIUS-Server Communication

RADIUS security servers are identified by their hostname or IP address, hostname and specific UDP port numbers, or IP address and specific UDP port numbers. The combination of the IP address and UDP port number creates a unique identifier, which enables RADIUS requests to be sent to multiple UDP ports on a server at the same IP address. If two different host entries on the same RADIUS server are configured for the same service—for example, authentication—the second host entry configured acts as the fail-over backup to the first one. The RADIUS host entries are tried in the order that they were configured.

802.1x Authentication with VLAN Assignment

The switch supports 802.1x authentication with VLAN assignment. After successful 802.1x authentication of a port, the RADIUS server sends the VLAN assignment to configure the switch port. The RADIUS server database maintains the username-to-VLAN mappings, assigning the VLAN based on the username of the client connected to the switch port. You can use this feature to limit network access for certain users.

Voice device authentication is supported with multidomain host mode in Cisco IOS Release 12.2(37)SE. In Cisco IOS Release 12.2(40)SE and later, when a voice device is authorized and the RADIUS server returned an authorized VLAN, the voice VLAN on the port is configured to send and receive packets on the assigned voice VLAN. Voice VLAN assignment behaves the same as data VLAN assignment on multidomain authentication (MDA)-enabled ports.

When configured on the switch and the RADIUS server, 802.1x authentication with VLAN assignment has these characteristics:

- If no VLAN is supplied by the RADIUS server or if 802.1x authentication is disabled, the port is configured in its access VLAN after successful authentication. Recall that an access VLAN is a VLAN assigned to an access port. All packets sent from or received on this port belong to this VLAN.
- If 802.1x authentication is enabled but the VLAN information from the RADIUS server is not valid, authorization fails and configured VLAN remains in use. This prevents ports from appearing unexpectedly in an inappropriate VLAN because of a configuration error.

Configuration errors could include specifying a VLAN for a routed port, a malformed VLAN ID, a nonexistent or internal (routed port) VLAN ID, an RSPAN VLAN, a shut down or suspended VLAN. In the case of a multidomain host port, configuration errors can also be due to an attempted assignment of a data VLAN that matches the configured or assigned voice VLAN ID (or the reverse).

- If 802.1x authentication is enabled and all information from the RADIUS server is valid, the authorized device is placed in the specified VLAN after authentication.
- If the multiple-hosts mode is enabled on an 802.1x port, all hosts are placed in the same VLAN (specified by the RADIUS server) as the first authenticated host.
- Enabling port security does not impact the RADIUS server-assigned VLAN behavior.
- If 802.1x authentication is disabled on the port, it is returned to the configured access VLAN and configured voice VLAN.
- If an 802.1x port is authenticated and put in the RADIUS server-assigned VLAN, any change to the port access VLAN configuration does not take effect. In the case of a multidomain host, the same applies to voice devices when the port is fully authorized with these exceptions:
 - If the VLAN configuration change of one device results in matching the other device configured or assigned VLAN, then authorization of all devices on the port is terminated and multidomain host mode is disabled until a valid configuration is restored where data and voice device configured VLANs no longer match.
 - If a voice device is authorized and is using a downloaded voice VLAN, the removal of the voice VLAN configuration, or modifying the configuration value to *dot1p* or *untagged* results in voice device un-authorization and the disablement of multi-domain host mode.

When the port is in the force authorized, force unauthorized, unauthorized, or shutdown state, it is put into the configured access VLAN.

To configure VLAN assignment you need to perform these tasks:

- Enable AAA authorization by using the **network** keyword to allow interface configuration from the RADIUS server.
- Enable 802.1x authentication. (The VLAN assignment feature is automatically enabled when you configure 802.1x authentication on an access port).
- Assign vendor-specific tunnel attributes in the RADIUS server. The RADIUS server must return these attributes to the switch:
 - [64] Tunnel-Type = VLAN
 - [65] Tunnel-Medium-Type = 802
 - [81] Tunnel-Private-Group-ID = VLAN name or VLAN ID
 - [83] Tunnel-Preference

Attribute [64] must contain the value *VLAN* (type 13). Attribute [65] must contain the value *802* (type 6). Attribute [81] specifies the *VLAN name* or *VLAN ID* assigned to the IEEE 802.1x-authenticated user.

802.1x Authentication with Per-User ACLs

You can enable per-user access control lists (ACLs) to provide different levels of network access and service to an 802.1x-authenticated user. When the RADIUS server authenticates a user connected to an 802.1x port, it retrieves the ACL attributes based on the user identity and sends them to the switch. The switch applies the attributes to the 802.1x port for the duration of the user session. The switch removes the per-user ACL configuration when the session is over, if authentication fails, or if a link-down condition occurs. The switch does not save RADIUS-specified ACLs in the running configuration. When the port is unauthorized, the switch removes the ACL from the port.

You can configure router ACLs and input port ACLs on the same switch. However, a port ACL takes precedence over a router ACL. If you apply input port ACL to an interface that belongs to a VLAN, the port ACL takes precedence over an input router ACL applied to the VLAN interface. Incoming packets received on the port, to which a port ACL is applied, are filtered by the port ACL. Incoming routed packets received on other ports are filtered by the router ACL. Outgoing routed packets are filtered by the router ACL. To avoid configuration conflicts, you should carefully plan the user profiles stored on the RADIUS server.

RADIUS supports per-user attributes, including vendor-specific attributes. These vendor-specific attributes (VSAs) are in octet-string format and are passed to the switch during the authentication process. The VSAs used for per-user ACLs are `inac1#<n>` for the ingress direction and `outacl#<n>` for the egress direction. MAC ACLs are supported only in the ingress direction. The switch supports VSAs only in the ingress direction. It does not support port ACLs in the egress direction on Layer 2 ports.

Use only the extended ACL syntax style to define the per-user configuration stored on the RADIUS server. When the definitions are passed from the RADIUS server, they are created by using the extended naming convention. However, if you use the Filter-Id attribute, it can point to a standard ACL.

You can use the Filter-Id attribute to specify an inbound or outbound ACL that is already configured on the switch. The attribute contains the ACL number followed by `.in` for ingress filtering or `.out` for egress filtering. If the RADIUS server does not allow the `.in` or `.out` syntax, the access list is applied to the outbound ACL by default. The user is marked unauthorized if the Filter-Id sent from the RADIUS server is not configured on the device. Because of limited support of Cisco IOS access lists on the switch, the Filter-Id attribute is supported only for IP ACLs numbered in the range of 1 to 199 (IP standard ACLs) and 1300 to 2699 (IP extended ACLs).

The maximum size of the per-user ACL is 4000 ASCII characters but is limited by the maximum size of RADIUS-server per-user ACLs.

You must meet the following prerequisites to configure per-user ACLs:

- Enable AAA authentication.
- Enable AAA authorization by using the **network** keyword to allow interface configuration from the RADIUS server.
- Enable 802.1x authentication.
- Configure the user profile and VSAs on the RADIUS server.
- Configure the 802.1x port for single-host mode.



Note Per-user ACLs are supported only in single-host mode.

802.1x Authentication with Downloadable ACLs and Redirect URLs



Note IPv6 does not support Redirect URLs.

You can download ACLs and redirect URLs from a RADIUS server to the switch during 802.1x authentication or MAC authentication bypass of the host. You can also download ACLs during web authentication.



Note A downloadable ACL is also referred to as a *dACL*.

If more than one host is authenticated and the host is in single-host, MDA, or multiple-authentication mode, the switch changes the source address of the ACL to the host IP address.

You can apply the ACLs and redirect URLs to all the devices connected to the 802.1x-enabled port.

If no ACLs are downloaded during 802.1x authentication, the switch applies the static default ACL on the port to the host. On a voice VLAN port configured in multi-auth or MDA mode, the switch applies the ACL only to the phone as part of the authorization policies.



Note The limit for dACL with stacking is 64 ACEs per dACL per port. The limit without stacking is the number of available TCAM entries which varies based on the other ACL features that are active.

If there is no static ACL on a port, a dynamic auth-default ACL is created, and policies are enforced before dACLs are downloaded and applied.



Note The auth-default-ACL does not appear in the running configuration.

The auth-default ACL is created when at least one host with an authorization policy is detected on the port. The auth-default ACL is removed from the port when the last authenticated session ends. You can configure the auth-default ACL for IPv4 by using the **ip access-list extended auth-default-acl** command in global configuration mode. For IPv6, use the **ipv6 access-list extended auth-default-acl** command in the global configuration mode.



Note The auth-default-ACL does not support Cisco Discovery Protocol bypass in the single host mode. You must configure a static ACL on the interface to support Cisco Discovery Protocol bypass.

The 802.1x and MAB authentication methods support two authentication modes, *open* and *closed*. If there is no static ACL on a port in *closed* authentication mode:

- An auth-default-ACL is created.
- The auth-default-ACL allows only DHCP traffic until policies are enforced.
- When the first host authenticates, the authorization policy is applied without IP address insertion.

- When a second host is detected, the policies for the first host are refreshed, and policies for the first and subsequent sessions are enforced with IP address insertion.

If there is no static ACL on a port in *open* authentication mode:

- An auth-default-ACL-OPEN is created and allows all traffic.
- Policies are enforced with IP address insertion to prevent security breaches.
- Web authentication is subject to the auth-default-ACL-OPEN.

To control access for hosts with no authorization policy, you can configure a directive. The supported values for the directive are *open* and *default*. When you configure the *open* directive, all traffic is allowed. The *default* directive subjects traffic to the access provided by the port. You can configure the directive either in the user profile on the AAA server or on the switch. To configure the directive on the AAA server, use the **authz-directive =<open/default>** global command. To configure the directive on the switch, use the **epm access-control open** global configuration command.



Note The default value of the directive is *default*.

If a host falls back to web authentication on a port without a configured ACL:

- If the port is in open authentication mode, the auth-default-ACL-OPEN is created.
- If the port is in closed authentication mode, the auth-default-ACL is created.

The access control entries (ACEs) in the fallback ACL are converted to per-user entries. If the configured fallback profile does not include a fallback ACL, the host is subject to the auth-default-ACL associated with the port.



Note If you use a custom logo with web authentication and it is stored on an external server, the port ACL must allow access to the external server before authentication. You must either configure a static port ACL or change the auth-default-ACL to provide appropriate access to the external server.

Cisco Secure ACS and Attribute-Value Pairs for the Redirect URL

The switch uses these *cisco-av-pair* VSAs:

- url-redirect is the HTTP or HTTPS URL.
- url-redirect-acl is the switch ACL name or number.

The switch uses the CiscoSecure-defined-ACL attribute value pair to intercept an HTTP or HTTPS request from the end point. The switch then forwards the client web browser to the specified redirect address. The url-redirect AV pair on the Cisco Secure ACS contains the URL to which the web browser is redirected. The url-redirect-acl attribute value pair contains the name or number of an ACL that specifies the HTTP or HTTPS traffic to redirect.

**Note**

- Traffic that matches a permit ACE in the ACL is redirected.
- Define the URL redirect ACL and the default port ACL on the switch.

If a redirect URL is configured for a client on the authentication server, a default port ACL on the connected client switch port must also be configured.

Cisco Secure ACS and Attribute-Value Pairs for Downloadable ACLs

You can set the CiscoSecure-Defined-ACL Attribute-Value (AV) pair on the Cisco Secure ACS with the RADIUS cisco-av-pair vendor-specific attributes (VSAs). This pair specifies the names of the downloadable ACLs on the Cisco Secure ACS with the #ACL#-IP-name-number attribute for IPv4 and #ACL#-.in.ipv6 attribute for IPv6.

- The *name* is the ACL name.
- The *number* is the version number (for example, 3f783768).

If a downloadable ACL is configured for a client on the authentication server, a default port ACL on the connected client switch port must also be configured.

If the default ACL is configured on the switch and the Cisco Secure ACS sends a host-access-policy to the switch, it applies the policy to traffic from the host connected to a switch port. If the policy does not apply, the switch applies the default ACL. If the Cisco Secure ACS sends the switch a downloadable ACL, this ACL takes precedence over the default ACL that is configured on the switch port. However, if the switch receives an host access policy from the Cisco Secure ACS but the default ACL is not configured, the authorization failure is declared.

VLAN ID-Based MAC Authentication

You can use VLAN ID-based MAC authentication if you wish to authenticate hosts based on a static VLAN ID instead of a downloadable VLAN. When you have a static VLAN policy configured on your switch, VLAN information is sent to an IAS (Microsoft) RADIUS server along with the MAC address of each host for authentication. The VLAN ID configured on the connected port is used for MAC authentication. By using VLAN ID-based MAC authentication with an IAS server, you can have a fixed number of VLANs in the network.

The feature also limits the number of VLANs monitored and handled by STP. The network can be managed as a fixed VLAN.

802.1x Authentication with Guest VLAN

You can configure a guest VLAN for each 802.1x port on the switch to provide limited services to clients, such as downloading the 802.1x client. These clients might be upgrading their system for 802.1x authentication, and some hosts, such as Windows 98 systems, might not be IEEE 802.1x-capable.

When you enable a guest VLAN on an 802.1x port, the switch assigns clients to a guest VLAN when the switch does not receive a response to its EAP request/identity frame or when EAPOL packets are not sent by the client.

The switch maintains the EAPOL packet history. If an EAPOL packet is detected on the interface during the lifetime of the link, the switch determines that the device connected to that interface is an IEEE 802.1x-capable supplicant, and the interface does not change to the guest VLAN state. EAPOL history is cleared if the interface link status goes down. If no EAPOL packet is detected on the interface, the interface changes to the guest VLAN state.

If the switch is trying to authorize an 802.1x-capable voice device and the AAA server is unavailable, the authorization attempt fails, but the detection of the EAPOL packet is saved in the EAPOL history. When the AAA server becomes available, the switch authorizes the voice device. However, the switch no longer allows other devices access to the guest VLAN. To prevent this situation, use one of these command sequences:

- Enter the **authentication event no-response action authorize vlan** *vlan-id* interface configuration command to allow access to the guest VLAN.
- Enter the **shutdown** interface configuration command followed by the **no shutdown** interface configuration command to restart the port.

If devices send EAPOL packets to the switch during the lifetime of the link, the switch no longer allows clients that fail authentication access to the guest VLAN.



Note If an EAPOL packet is detected after the interface has changed to the guest VLAN, the interface reverts to an unauthorized state, and 802.1x authentication restarts.

Any number of 802.1x-incapable clients are allowed access when the switch port is moved to the guest VLAN. If an 802.1x-capable client joins the same port on which the guest VLAN is configured, the port is put into the unauthorized state in the user-configured access VLAN, and authentication is restarted.

Guest VLANs are supported on 802.1x ports in single host, multiple host, multi-auth and multi-domain modes.

You can configure any active VLAN except an RSPAN VLAN, a private VLAN, or a voice VLAN as an 802.1x guest VLAN. The guest VLAN feature is not supported on internal VLANs (routed ports) or trunk ports; it is supported only on access ports.

The switch supports *MAC authentication bypass*. When MAC authentication bypass is enabled on an 802.1x port, the switch can authorize clients based on the client MAC address when IEEE 802.1x authentication times out while waiting for an EAPOL message exchange. After detecting a client on an 802.1x port, the switch waits for an Ethernet packet from the client. The switch sends the authentication server a RADIUS-access/request frame with a username and password based on the MAC address. If authorization succeeds, the switch grants the client access to the network. If authorization fails, the switch assigns the port to the guest VLAN if one is specified.

802.1x Authentication with Restricted VLAN

You can configure a restricted VLAN (also referred to as an *authentication failed VLAN*) for each IEEE 802.1x port on a switch stack or a switch to provide limited services to clients that cannot access the guest VLAN. These clients are 802.1x-compliant and cannot access another VLAN because they fail the authentication process. A restricted VLAN allows users without valid credentials in an authentication server (typically, visitors to an enterprise) to access a limited set of services. The administrator can control the services available to the restricted VLAN.



Note You can configure a VLAN to be both the guest VLAN and the restricted VLAN if you want to provide the same services to both types of users.

Without this feature, the client attempts and fails authentication indefinitely, and the switch port remains in the spanning-tree blocking state. With this feature, you can configure the switch port to be in the restricted VLAN after a specified number of authentication attempts (the default value is 3 attempts).

The authenticator counts the failed authentication attempts for the client. When this count exceeds the configured maximum number of authentication attempts, the port moves to the restricted VLAN. The failed attempt count increments when the RADIUS server replies with either an *EAP failure* or an empty response without an EAP packet. When the port moves into the restricted VLAN, the failed attempt counter resets.

Users who fail authentication remain in the restricted VLAN until the next re-authentication attempt. A port in the restricted VLAN tries to re-authenticate at configured intervals (the default is 60 seconds). If re-authentication fails, the port remains in the restricted VLAN. If re-authentication is successful, the port moves either to the configured VLAN or to a VLAN sent by the RADIUS server. You can disable re-authentication. If you do this, the only way to restart the authentication process is for the port to receive a *link down* or *EAP logoff* event. We recommend that you keep re-authentication enabled if a client might connect through a hub. When a client disconnects from the hub, the port might not receive the *link down* or *EAP logoff* event.

After a port moves to the restricted VLAN, a simulated EAP success message is sent to the client. This prevents clients from indefinitely attempting authentication. Some clients (for example, devices running Windows XP) cannot implement DHCP without EAP success.

Restricted VLANs are supported on 802.1x ports in all host modes and on Layer 2 ports.

You can configure any active VLAN except an RSPAN VLAN, a primary private VLAN, or a voice VLAN as an 802.1x restricted VLAN. The restricted VLAN feature is not supported on internal VLANs (routed ports) or trunk ports; it is supported only on access ports.

Other security port features such as dynamic ARP Inspection, DHCP snooping, and IP source guard can be configured independently on a restricted VLAN.

802.1x Authentication with Inaccessible Authentication Bypass

Use the inaccessible authentication bypass feature, also referred to as *critical authentication* or the *AAA fail policy*, when the switch cannot reach the configured RADIUS servers and new hosts cannot be authenticated. You can configure the switch to connect those hosts to *critical ports*.

When a new host tries to connect to the critical port, that host is moved to a user-specified access VLAN, the *critical VLAN*. The administrator gives limited authentication to the hosts.

When the switch tries to authenticate a host connected to a critical port, the switch checks the status of the configured RADIUS server. If a server is available, the switch can authenticate the host. However, if all the RADIUS servers are unavailable, the switch grants network access to the host and puts the port in the *critical-authentication* state, which is a special case of the authentication state.



Note If *critical authentication* is configured on interface, then *vlan* used for critical authorization (*critical vlan*) should be active on the switch. If the *critical vlan* is inactive (or) down, *critical authentication* session will keep trying to enable inactive *vlan* and fail repeatedly. This can lead to large amount of memory holding.

Inaccessible Authentication Bypass Support on Multiple-Authentication Ports

When a port is configured on any host mode and the AAA server is unavailable, the port is then configured to multi-host mode and moved to the critical VLAN. To support this inaccessible bypass on multiple-authentication (multiauth) ports, use the **authentication event server dead action reinitialize vlan *vlan-id*** command. When a new host tries to connect to the critical port, that port is reinitialized and all the connected hosts are moved to the user-specified access VLAN.

This command is supported on all host modes.

Inaccessible Authentication Bypass Authentication Results

The behavior of the inaccessible authentication bypass feature depends on the authorization state of the port:

- If the port is unauthorized when a host connected to a critical port tries to authenticate and all servers are unavailable, the switch puts the port in the critical-authentication state in the RADIUS-configured or user-specified access VLAN.
- If the port is already authorized and reauthentication occurs, the switch puts the critical port in the critical-authentication state in the current VLAN, which might be the one previously assigned by the RADIUS server.
- If the RADIUS server becomes unavailable during an authentication exchange, the current exchange times out, and the switch puts the critical port in the critical-authentication state during the next authentication attempt.

You can configure the critical port to reinitialize hosts and move them out of the critical VLAN when the RADIUS server is again available. When this is configured, all critical ports in the critical-authentication state are automatically re-authenticated.

Inaccessible Authentication Bypass Feature Interactions

Inaccessible authentication bypass interacts with these features:

- Guest VLAN—Inaccessible authentication bypass is compatible with guest VLAN. When a guest VLAN is enabled on 8021.x port, the features interact as follows:
 - If at least one RADIUS server is available, the switch assigns a client to a guest VLAN when the switch does not receive a response to its EAP request/identity frame or when EAPOL packets are not sent by the client.
 - If all the RADIUS servers are not available and the client is connected to a critical port, the switch authenticates the client and puts the critical port in the critical-authentication state in the RADIUS-configured or user-specified access VLAN.
 - If all the RADIUS servers are not available and the client is not connected to a critical port, the switch might not assign clients to the guest VLAN if one is configured.

- If all the RADIUS servers are not available and if a client is connected to a critical port and was previously assigned to a guest VLAN, the switch keeps the port in the guest VLAN.
- Restricted VLAN—If the port is already authorized in a restricted VLAN and the RADIUS servers are unavailable, the switch puts the critical port in the critical-authentication state in the restricted VLAN.
- 802.1x accounting—Accounting is not affected if the RADIUS servers are unavailable.
- Private VLAN—You can configure inaccessible authentication bypass on a private VLAN host port. The access VLAN must be a secondary private VLAN.
- Voice VLAN—Inaccessible authentication bypass is compatible with voice VLAN, but the RADIUS-configured or user-specified access VLAN and the voice VLAN must be different.
- Remote Switched Port Analyzer (RSPAN)—Do not configure an RSPAN VLAN as the RADIUS-configured or user-specified access VLAN for inaccessible authentication bypass.

In a switch stack:

- The stack's active switch checks the status of the RADIUS servers by sending keepalive packets. When the status of a RADIUS server changes, the stack's active switch sends the information to the stack members. The stack members can then check the status of RADIUS servers when re-authenticating critical ports.
- If the new active switch is elected, the link between the switch stack and RADIUS server might change, and the new stack immediately sends keepalive packets to update the status of the RADIUS servers. If the server status changes from *dead* to *alive*, the switch re-authenticates all switch ports in the critical-authentication state.

When a member is added to the stack, the stack's active switch sends the member the server status.

802.1x Critical Voice VLAN

When an IP phone connected to a port is authenticated by the Cisco Identity Services Engine (ISE), the phone is put into the voice domain. If the ISE is not reachable, the switch cannot determine if the device is a voice device. If the server is unavailable, the phone cannot access the voice network and therefore cannot operate.

For data traffic, you can configure inaccessible authentication bypass, or critical authentication, to allow traffic to pass through on the native VLAN when the server is not available. If the RADIUS authentication server is unavailable (down) and inaccessible authentication bypass is enabled, the switch grants the client access to the network and puts the port in the critical-authentication state in the RADIUS-configured or the user-specified access VLAN. When the switch cannot reach the configured RADIUS servers and new hosts cannot be authenticated, the switch connects those hosts to critical ports. A new host trying to connect to the critical port is moved to a user-specified access VLAN, the critical VLAN, and granted limited authentication.

You can enter the **authentication event server dead action authorize voice** interface configuration command to configure the critical voice VLAN feature. When the ISE does not respond, the port goes into critical authentication mode. When traffic coming from the host is tagged with the voice VLAN, the connected device (the phone) is put in the configured voice VLAN for the port. The IP phones learn the voice VLAN identification through Cisco Discovery Protocol (Cisco devices) or through LLDP or DHCP.

You can configure the voice VLAN for a port by entering the **switchport voice vlan *vlan-id*** interface configuration command.

This feature is supported in multidomain and multi-auth host modes. Although you can enter the command when the switch is in single-host or multi-host mode, the command has no effect unless the device changes to multidomain or multi-auth host mode.

802.1x User Distribution

You can configure 802.1x user distribution to load-balance users with the same group name across multiple different VLANs.

The VLANs are either supplied by the RADIUS server or configured through the switch CLI under a VLAN group name.

- Configure the RADIUS server to send more than one VLAN name for a user. The multiple VLAN names can be sent as part of the response to the user. The 802.1x user distribution tracks all the users in a particular VLAN and achieves load balancing by moving the authorized user to the least populated VLAN.
- Configure the RADIUS server to send a VLAN group name for a user. The VLAN group name can be sent as part of the response to the user. You can search for the selected VLAN group name among the VLAN group names that you configured by using the switch CLI. If the VLAN group name is found, the corresponding VLANs under this VLAN group name are searched to find the least populated VLAN. Load balancing is achieved by moving the corresponding authorized user to that VLAN.



Note The RADIUS server can send the VLAN information in any combination of VLAN-IDs, VLAN names, or VLAN groups.

802.1x User Distribution Configuration Guidelines

- Confirm that at least one VLAN is mapped to the VLAN group.
- You can map more than one VLAN to a VLAN group.
- You can modify the VLAN group by adding or deleting a VLAN.
- When you clear an existing VLAN from the VLAN group name, none of the authenticated ports in the VLAN are cleared, but the mappings are removed from the existing VLAN group.
- If you clear the last VLAN from the VLAN group name, the VLAN group is cleared.
- You can clear a VLAN group even when the active VLANs are mapped to the group. When you clear a VLAN group, none of the ports or users that are in the authenticated state in any VLAN within the group are cleared, but the VLAN mappings to the VLAN group are cleared.

IEEE 802.1x Authentication with Voice VLAN Ports

A voice VLAN port is a special access port associated with two VLAN identifiers:

- VVID to carry voice traffic to and from the IP phone. The VVID is used to configure the IP phone connected to the port.

- PVID to carry the data traffic to and from the workstation connected to the switch through the IP phone. The PVID is the native VLAN of the port.

The IP phone uses the VVID for its voice traffic, regardless of the authorization state of the port. This allows the phone to work independently of IEEE 802.1x authentication.

In single-host mode, only the IP phone is allowed on the voice VLAN. In multiple-hosts mode, additional clients can send traffic on the voice VLAN after a supplicant is authenticated on the PVID. When multiple-hosts mode is enabled, the supplicant authentication affects both the PVID and the VVID.

A voice VLAN port becomes active when there is a link, and the device MAC address appears after the first CDP message from the IP phone. Cisco IP phones do not relay CDP messages from other devices. As a result, if several IP phones are connected in series, the switch recognizes only the one directly connected to it. When IEEE 802.1x authentication is enabled on a voice VLAN port, the switch drops packets from unrecognized IP phones more than one hop away.

When IEEE 802.1x authentication is enabled on a switch port, you can configure an access port VLAN that is also a voice VLAN.

When IP phones are connected to an 802.1x-enabled switch port that is in single host mode, the switch grants the phones network access without authenticating them. We recommend that you use multidomain authentication (MDA) on the port to authenticate both a data device and a voice device, such as an IP phone



Note If you enable IEEE 802.1x authentication on an access port on which a voice VLAN is configured and to which a Cisco IP Phone is connected, the Cisco IP phone loses connectivity to the switch for up to 30 seconds.

IEEE 802.1x Authentication with Port Security

In general, Cisco does not recommend enabling port security when IEEE 802.1x is enabled. Since IEEE 802.1x enforces a single MAC address per port (or per VLAN when MDA is configured for IP telephony), port security is redundant and in some cases may interfere with expected IEEE 802.1x operations.

IEEE 802.1x Authentication with Wake-on-LAN

The IEEE 802.1x authentication with wake-on-LAN (WoL) feature allows dormant PCs to be powered when the switch receives a specific Ethernet frame, known as the *magic packet*. You can use this feature in environments where administrators need to connect to systems that have been powered down.

When a host that uses WoL is attached through an IEEE 802.1x port and the host powers off, the IEEE 802.1x port becomes unauthorized. The port can only receive and send EAPOL packets, and WoL magic packets cannot reach the host. When the PC is powered off, it is not authorized, and the switch port is not opened.

When the switch uses IEEE 802.1x authentication with WoL, the switch forwards traffic to unauthorized IEEE 802.1x ports, including magic packets. While the port is unauthorized, the switch continues to block ingress traffic other than EAPOL packets. The host can receive packets but cannot send packets to other devices in the network.



Note If PortFast is not enabled on the port, the port is forced to the bidirectional state.

When you configure a port as unidirectional by using the **authentication control-direction in** interface configuration command, the port changes to the spanning-tree forwarding state. The port can send packets to the host but cannot receive packets from the host.

When you configure a port as bidirectional by using the **authentication control-direction both** interface configuration command, the port is access-controlled in both directions. The port does not receive packets from or send packets to the host.

IEEE 802.1x Authentication with MAC Authentication Bypass

You can configure the switch to authorize clients based on the client MAC address by using the MAC authentication bypass feature. For example, you can enable this feature on IEEE 802.1x ports connected to devices such as printers.

If IEEE 802.1x authentication times out while waiting for an EAPOL response from the client, the switch tries to authorize the client by using MAC authentication bypass.

When the MAC authentication bypass feature is enabled on an IEEE 802.1x port, the switch uses the MAC address as the client identity. The authentication server has a database of client MAC addresses that are allowed network access. After detecting a client on an IEEE 802.1x port, the switch waits for an Ethernet packet from the client. The switch sends the authentication server a RADIUS-access/request frame with a username and password based on the MAC address. If authorization succeeds, the switch grants the client access to the network. If authorization fails, the switch assigns the port to the guest VLAN if one is configured. This process works for most client devices; however, it does not work for clients that use an alternate MAC address format. You can configure how MAB authentication is performed for clients with MAC addresses that deviate from the standard format or where the RADIUS configuration requires the user name and password to differ.

If an EAPOL packet is detected on the interface during the lifetime of the link, the switch determines that the device connected to that interface is an 802.1x-capable supplicant and uses 802.1x authentication (not MAC authentication bypass) to authorize the interface. EAPOL history is cleared if the interface link status goes down.

If the switch already authorized a port by using MAC authentication bypass and detects an IEEE 802.1x supplicant, the switch does not unauthorize the client connected to the port. When re-authentication occurs, the switch uses the authentication or re-authentication methods configured on the port, if the previous session ended because the Termination-Action RADIUS attribute value is `DEFAULT`.

Clients that were authorized with MAC authentication bypass can be re-authenticated. The re-authentication process is the same as that for clients that were authenticated with IEEE 802.1x. During re-authentication, the port remains in the previously assigned VLAN. If re-authentication is successful, the switch keeps the port in the same VLAN. If re-authentication fails, the switch assigns the port to the guest VLAN, if one is configured.

If re-authentication is based on the Session-Timeout RADIUS attribute (Attribute[27]) and the Termination-Action RADIUS attribute (Attribute [29]) and if the Termination-Action RADIUS attribute (Attribute [29]) action is *Initialize* (the attribute value is `DEFAULT`), the MAC authentication bypass session ends, and connectivity is lost during re-authentication. If MAC authentication bypass is enabled and the IEEE 802.1x authentication times out, the switch uses the MAC authentication bypass feature to initiate re-authorization. For more information about these AV pairs, see RFC 3580, "IEEE 802.1X Remote Authentication Dial In User Service (RADIUS) Usage Guidelines."

MAC authentication bypass interacts with the features:

- IEEE 802.1x authentication—You can enable MAC authentication bypass only if 802.1x authentication is enabled on the port .

- Guest VLAN—If a client has an invalid MAC address identity, the switch assigns the client to a guest VLAN if one is configured.
- Restricted VLAN—This feature is not supported when the client connected to an IEEE 802.1x port is authenticated with MAC authentication bypass.
- Port security
- Voice VLAN
- Private VLAN—You can assign a client to a private VLAN.

Cisco IOS Release 12.2(55)SE and later supports filtering of verbose MAB system messages

Network Admission Control Layer 2 IEEE 802.1x Validation

The switch supports the Network Admission Control (NAC) Layer 2 IEEE 802.1x validation, which checks the antivirus condition or *posture* of endpoint systems or clients before granting the devices network access. With NAC Layer 2 IEEE 802.1x validation, you can do these tasks:

- Download the Session-Timeout RADIUS attribute (Attribute[27]) and the Termination-Action RADIUS attribute (Attribute[29]) from the authentication server.
- Set the number of seconds between re-authentication attempts as the value of the Session-Timeout RADIUS attribute (Attribute[27]) and get an access policy against the client from the RADIUS server.
- Set the action to be taken when the switch tries to re-authenticate the client by using the Termination-Action RADIUS attribute (Attribute[29]). If the value is the *DEFAULT* or is not set, the session ends. If the value is RADIUS-Request, the re-authentication process starts.
- Set the list of VLAN number or name or VLAN group name as the value of the Tunnel Group Private ID (Attribute[81]) and the preference for the VLAN number or name or VLAN group name as the value of the Tunnel Preference (Attribute[83]). If you do not configure the Tunnel Preference, the first Tunnel Group Private ID (Attribute[81]) attribute is picked up from the list.
- View the NAC posture token, which shows the posture of the client, by using the **show authentication** privileged EXEC command.
- Configure secondary private VLANs as guest VLANs.

Configuring NAC Layer 2 IEEE 802.1x validation is similar to configuring IEEE 802.1x port-based authentication except that you must configure a posture token on the RADIUS server.

Flexible Authentication Ordering

You can use flexible authentication ordering to configure the order of methods that a port uses to authenticate a new host. The IEEE 802.1X Flexible Authentication feature supports three authentication methods:

- dot1X—IEEE 802.1X authentication is a Layer 2 authentication method.
- mab—MAC-Authentication Bypass is a Layer 2 authentication method.
- webauth—Web authentication is a Layer 3 authentication method.

Using this feature, you can control which ports use which authentication methods, and you can control the failover sequencing of methods on those ports. For example, MAC authentication bypass and 802.1x can be the primary or secondary authentication methods, and web authentication can be the fallback method if either or both of those authentication attempts fail.

The IEEE 802.1X Flexible Authentication feature supports the following host modes:

- multi-auth—Multiauthentication allows one authentication on a voice VLAN and multiple authentications on the data VLAN.
- multi-domain—Multidomain authentication allows two authentications: one on the voice VLAN and one on the data VLAN.

Open1x Authentication

Open1x authentication allows a device access to a port before that device is authenticated. When open authentication is configured, a new host can pass traffic according to the access control list (ACL) defined on the port. After the host is authenticated, the policies configured on the RADIUS server are applied to that host.

You can configure open authentication with these scenarios:

- Single-host mode with open authentication—Only one user is allowed network access before and after authentication.
- MDA mode with open authentication—Only one user in the voice domain and one user in the data domain are allowed.
- Multiple-hosts mode with open authentication—Any host can access the network.
- Multiple-authentication mode with open authentication—Similar to MDA, except multiple hosts can be authenticated.



Note If open authentication is configured, it takes precedence over other authentication controls. This means that if you use the **authentication open** interface configuration command, the port will grant access to the host irrespective of the **authentication port-control** interface configuration command.

Multidomain Authentication

The switch supports multidomain authentication (MDA), which allows both a data device and voice device, such as an IP phone (Cisco or non-Cisco), to authenticate on the same switch port. The port is divided into a data domain and a voice domain.



Note For all host modes, the line protocol stays up before authorization when port-based authentication is configured.

MDA does not enforce the order of device authentication. However, for best results, we recommend that a voice device is authenticated before a data device on an MDA-enabled port.



Note When migrating from Cisco Discovery Protocol bypass to next-generation authentication bypass, if single or multi-host mode is used with an IP phone and one or more data devices, then move to multi-authentication mode with next-generation authentication bypass that provides the session visibility advantage.

Follow these guidelines for configuring MDA:

- You must configure a switch port for MDA.
- You must configure the voice VLAN for the IP phone when the host mode is set to multidomain.
- Voice VLAN assignment on an MDA-enabled port is supported Cisco IOS Release 12.2(40)SE and later.
- To authorize a voice device, the AAA server must be configured to send a Cisco Attribute-Value (AV) pair attribute with a value of *device-traffic-class=voice*. Without this value, the switch treats the voice device as a data device.
- The guest VLAN and restricted VLAN features only apply to the data devices on an MDA-enabled port. The switch treats a voice device that fails authorization as a data device.
- If more than one device attempts authorization on either the voice or the data domain of a port, it is error disabled.
- Until a device is authorized, the port drops its traffic. Non-Cisco IP phones or voice devices are allowed into both the data and voice VLANs. The data VLAN allows the voice device to contact a DHCP server to obtain an IP address and acquire the voice VLAN information. After the voice device starts sending on the voice VLAN, its access to the data VLAN is blocked.
- A voice device MAC address that is binding on the data VLAN is not counted towards the port security MAC address limit.
- MDA can use MAC authentication bypass as a fallback mechanism to allow the switch port to connect to devices that do not support IEEE 802.1x authentication.
- When a *data* or a *voice* device is detected on a port, its MAC address is blocked until authorization succeeds. If the authorization fails, the MAC address remains blocked for 5 minutes.
- If more than five devices are detected on the *data* VLAN or more than one voice device is detected on the *voice* VLAN while a port is unauthorized, the port is error disabled.
- When a port host mode is changed from single- or multihost to multidomain mode, an authorized data device remains authorized on the port. However, a Cisco IP phone that has been allowed on the port voice VLAN is automatically removed and must be reauthenticated on that port.
- Active fallback mechanisms such as guest VLAN and restricted VLAN remain configured after a port changes from single- or multihost mode to multidomain mode.
- Switching a port host mode from multidomain to single- or multihost mode removes all authorized devices from the port.
- If a data domain is authorized first and placed in the guest VLAN, non-IEEE 802.1x-capable voice devices need to tag their packets on the voice VLAN to trigger authentication.
- We do not recommend per-user ACLs with an MDA-enabled port. An authorized device with a per-user ACL policy might impact traffic on both the voice and data VLANs of the port. If used, only one device on the port should enforce per-user ACLs.

Limiting Login for Users

The Limiting Login feature helps Network administrators to limit the login attempt of users to a network. When a user fails to successfully login to a network within a configurable number of attempts within a configurable time limit, the user can be blocked. This feature is enabled only for local users and not for remote users. You need to configure the **aaa authentication rejected** command in global configuration mode to enable this feature.

802.1x Supplicant and Authenticator Switches with Network Edge Access Topology (NEAT)

The Network Edge Access Topology (NEAT) feature extends identity to areas outside the wiring closet (such as conference rooms). This allows any type of device to authenticate on the port.

- 802.1x switch supplicant: You can configure a switch to act as a supplicant to another switch by using the 802.1x supplicant feature. This configuration is helpful in a scenario, where, for example, a switch is outside a wiring closet and is connected to an upstream switch through a trunk port. A switch configured with the 802.1x switch supplicant feature authenticates with the upstream switch for secure connectivity. Once the supplicant switch authenticates successfully the port mode changes from access to trunk in an authenticator switch. In a supplicant switch you must manually configure trunk when enabling CISP.



Note NEAT configuration is the only supported and qualified method to authenticate switches using 802.1x. Any other method to authenticate a network switch can result in an undefined behavior.

- If the access VLAN is configured on the authenticator switch, it becomes the native VLAN for the trunk port after successful authentication.

In the default state, when you connect a supplicant switch to an authenticator switch that has BPDU guard enabled, the authenticator port could be error-disabled if it receives a Spanning Tree Protocol (STP) bridge protocol data unit (BPDU) packets before the supplicant switch has authenticated. Beginning with Cisco IOS Release 15.0(1)SE, you can control traffic exiting the supplicant port during the authentication period. Entering the **dot1x supplicant controlled transient** global configuration command temporarily blocks the supplicant port during authentication to ensure that the authenticator port does not shut down before authentication completes. If authentication fails, the supplicant port opens. Entering the **no dot1x supplicant controlled transient** global configuration command opens the supplicant port during the authentication period. This is the default behavior.

We strongly recommend using the **dot1x supplicant controlled transient** command on a supplicant switch when BPDU guard is enabled on the authenticator switch port with the **spanning-tree bpduguard enable** interface configuration command.



Note If you globally enable BPDU guard on the authenticator switch by using the **spanning-tree portfast bpduguard default** global configuration command, entering the **dot1x supplicant controlled transient** command does not prevent the BPDU violation.

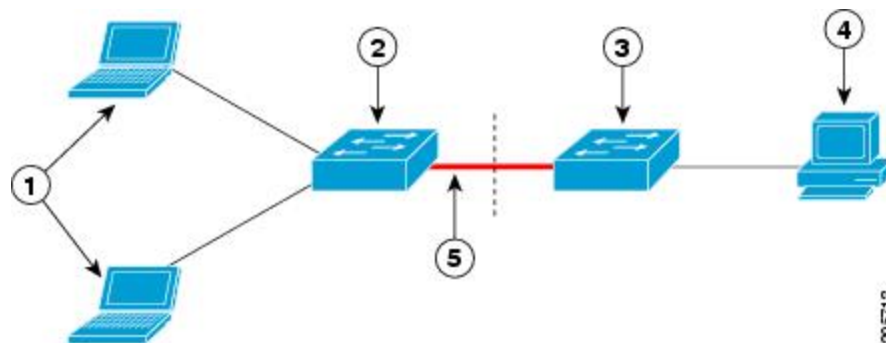
You can enable MDA or multiauth mode on the authenticator switch interface that connects to one more supplicant switches. Multihost mode is not supported on the authenticator switch interface.

When you reboot an authenticator switch with single-host mode enabled on the interface, the interface may move to err-disabled state before authentication. To recover from err-disabled state, flap the authenticator port to activate the interface again and initiate authentication.

Use the **dot1x supplicant force-multicast** global configuration command on the supplicant switch for Network Edge Access Topology (NEAT) to work in all host modes.

- **Host Authorization:** Ensures that only traffic from authorized hosts (connecting to the switch with supplicant) is allowed on the network. The switches use Client Information Signalling Protocol (CISP) to send the MAC addresses connecting to the supplicant switch to the authenticator switch.
- **Auto enablement:** Automatically enables trunk configuration on the authenticator switch, allowing user traffic from multiple VLANs coming from supplicant switches. Configure the `cisco-av-pair as device-traffic-class=switch` at the ISE. (You can configure this under the *group* or the *user* settings.)

Figure 113: Authenticator and Supplicant Switch using CISP



1	Workstations (clients)	2	Supplicant switch (outside wiring closet)
3	Authenticator switch	4	Cisco ISE
5	Trunk port		



Note The **switchport nonegotiate** command is not supported on supplicant and authenticator switches with NEAT. This command should not be configured at the supplicant side of the topology. If configured on the authenticator side, the internal macros will automatically remove this command from the port.

Voice Aware 802.1x Security



Note To use voice aware IEEE 802.1x authentication, the switch must be running the LAN base image.

You use the voice aware 802.1x security feature to configure the switch to disable only the VLAN on which a security violation occurs, whether it is a data or voice VLAN. In previous releases, when an attempt to authenticate the data client caused a security violation, the entire port shut down, resulting in a complete loss of connectivity.

You can use this feature in IP phone deployments where a PC is connected to the IP phone. A security violation found on the data VLAN results in the shutdown of only the data VLAN. The traffic on the voice VLAN flows through the switch without interruption.

Common Session ID

Authentication manager uses a single session ID (referred to as a common session ID) for a client no matter which authentication method is used. This ID is used for all reporting purposes, such as the show commands and MIBs. The session ID appears with all per-session syslog messages.

The session ID includes:

- The IP address of the Network Access Device (NAD)
- A monotonically increasing unique 32 bit integer
- The session start time stamp (a 32 bit integer)

This example shows how the session ID appears in the output of the show authentication command. The session ID in this example is 160000050000000B288508E5:

```
Device# show authentication sessions
Interface  MAC Address      Method  Domain  Status      Session ID
Fa4/0/4    0000.0000.0203  mab     DATA   Authz Success 160000050000000B288508E5
```

This is an example of how the session ID appears in the syslog output. The session ID in this example is also 160000050000000B288508E5:

```
1w0d: %AUTHMGR-5-START: Starting 'mab' for client (0000.0000.0203) on Interface Fa4/0/4
AuditSessionID 160000050000000B288508E5
1w0d: %MAB-5-SUCCESS: Authentication successful for client (0000.0000.0203) on Interface
Fa4/0/4 AuditSessionID 160000050000000B288508E5
1w0d: %AUTHMGR-7-RESULT: Authentication result 'success' from 'mab' for client
(0000.0000.0203) on Interface Fa4/0/4 AuditSessionID 160000050000000B288508E5
```

The session ID is used by the NAD, the AAA server, and other report-analyzing applications to identify the client. The ID appears automatically. No configuration is required.

How to Configure 802.1x Port-Based Authentication

Default 802.1x Authentication Configuration

Table 146: Default 802.1x Authentication Configuration

Feature	Default Setting
Switch 802.1x enable state	Disabled.
Per-port 802.1x enable state	Disabled (force-authorized). The port sends and receives normal traffic without 802.1x-based authentication of the client.
AAA	Disabled.
RADIUS server <ul style="list-style-type: none"> • IP address • UDP authentication port • Default accounting port • Key 	<ul style="list-style-type: none"> • None specified. • 1645. • 1646. • None specified.
Host mode	Single-host mode.
Control direction	Bidirectional control.
Periodic re-authentication	Disabled.
Number of seconds between re-authentication attempts	3600 seconds.
Re-authentication number	2 times (number of times that the switch restarts the authentication process before the port changes to the unauthorized state).
Quiet period	60 seconds (number of seconds that the switch remains in the quiet state following a failed authentication exchange with the client).
Retransmission time	30 seconds (number of seconds that the switch should wait for a response to an EAP request/identity frame from the client before resending the request).
Maximum retransmission number	2 times (number of times that the switch will send an EAP-request/identity frame before restarting the authentication process).

Feature	Default Setting
Client timeout period	30 seconds (when relaying a request from the authentication server to the client, the amount of time the switch waits for a response before resending the request to the client.)
Authentication server timeout period	30 seconds (when relaying a response from the client to the authentication server, the amount of time the switch waits for a reply before resending the response to the server.) You can change this timeout period by using the dot1x timeout server-timeout interface configuration command.
Inactivity timeout	Disabled.
Guest VLAN	None specified.
Inaccessible authentication bypass	Disabled.
Restricted VLAN	None specified.
Authenticator (switch) mode	None specified.
MAC authentication bypass	Disabled.
Voice-aware security	Disabled.

802.1x Authentication Configuration Guidelines

802.1x Authentication

These are the 802.1x authentication configuration guidelines:

- When 802.1x authentication is enabled, ports are authenticated before any other Layer 2 or Layer 3 features are enabled.
- If the VLAN to which an 802.1x-enabled port is assigned changes, this change is transparent and does not affect the switch. For example, this change occurs if a port is assigned to a RADIUS server-assigned VLAN and is then assigned to a different VLAN after re-authentication.

If the VLAN to which an 802.1x port is assigned to shut down, disabled, or removed, the port becomes unauthorized. For example, the port is unauthorized after the access VLAN to which a port is assigned shuts down or is removed.

- The 802.1x protocol is supported on Layer 2 static-access ports, voice VLAN ports, and Layer 3 routed ports, but it is not supported on these port types:
 - Dynamic ports—A port in dynamic mode can negotiate with its neighbor to become a trunk port. If you try to enable 802.1x authentication on a dynamic port, an error message appears, and 802.1x authentication is not enabled. If you try to change the mode of an 802.1x-enabled port to dynamic, an error message appears, and the port mode is not changed.

- EtherChannel port—Do not configure a port that is an active or a not-yet-active member of an EtherChannel as an 802.1x port. If you try to enable 802.1x authentication on an EtherChannel port, an error message appears, and 802.1x authentication is not enabled.
- Switched Port Analyzer (SPAN) and Remote SPAN (RSPAN) destination ports—You can enable 802.1x authentication on a port that is a SPAN or RSPAN destination port. However, 802.1x authentication is disabled until the port is removed as a SPAN or RSPAN destination port. You can enable 802.1x authentication on a SPAN or RSPAN source port.
- Before globally enabling 802.1x authentication on a switch by entering the **dot1x system-auth-control** global configuration command, remove the EtherChannel configuration from the interfaces on which 802.1x authentication and EtherChannel are configured.
- Cisco IOS Release 12.2(55)SE and later supports filtering of system messages related to 802.1x authentication.

VLAN Assignment, Guest VLAN, Restricted VLAN, and Inaccessible Authentication Bypass

These are the configuration guidelines for VLAN assignment, guest VLAN, restricted VLAN, and inaccessible authentication bypass:

- When 802.1x authentication is enabled on a port, you cannot configure a port VLAN that is equal to a voice VLAN.
- You can configure any VLAN except an RSPAN VLAN or a voice VLAN as an 802.1x guest VLAN. The guest VLAN feature is not supported on internal VLANs (routed ports) or trunk ports; it is supported only on access ports.
- After you configure a guest VLAN for an 802.1x port to which a DHCP client is connected, you might need to get a host IP address from a DHCP server. You can change the settings for restarting the 802.1x authentication process on the switch before the DHCP process on the client times out and tries to get a host IP address from the DHCP server. Decrease the settings for the 802.1x authentication process (**authentication timer inactivity** and **authentication timer reauthentication** interface configuration commands). The amount to decrease the settings depends on the connected 802.1x client type.
- When configuring the inaccessible authentication bypass feature, follow these guidelines:
 - The feature is supported on 802.1x port in single-host mode and multihosts mode.
 - If the client is running Windows XP and the port to which the client is connected is in the critical-authentication state, Windows XP might report that the interface is not authenticated.
 - If the Windows XP client is configured for DHCP and has an IP address from the DHCP server, receiving an EAP-Success message on a critical port might not re-initiate the DHCP configuration process.
 - You can configure the inaccessible authentication bypass feature and the restricted VLAN on an 802.1x port. If the switch tries to re-authenticate a critical port in a restricted VLAN and all the RADIUS servers are unavailable, switch changes the port state to the critical authentication state and remains in the restricted VLAN.
 - If the CTS links are in Critical Authentication mode and the active switch reloads, the policy where SGT was configured on a device will not be available on the new active switch. This is because the internal bindings will not be synced to the standby switch in a 3750-X switch stack.

- You can configure any VLAN except an RSPAN VLAN or a voice VLAN as an 802.1x restricted VLAN. The restricted VLAN feature is not supported on internal VLANs (routed ports) or trunk ports; it is supported only on access ports.
- When wireless guest clients obtains IP from foreign client VLAN instead of anchor client VLAN, you should use the **ip dhcp required** command under the WLAN configuration to force clients to issue a new DHCP request. This prevents the clients from getting an incorrect IP at anchor.
- If the wired guest clients fail to get IP address after a Cisco WLC (foreign) reload, perform a shut/no shut on the ports used by the clients to reconnect them.

MAC Authentication Bypass

These are the MAC authentication bypass configuration guidelines:

- Unless otherwise stated, the MAC authentication bypass guidelines are the same as the 802.1x authentication guidelines.
- If you disable MAC authentication bypass from a port after the port has been authorized with its MAC address, the port state is not affected.
- If the port is in the unauthorized state and the client MAC address is not the authentication-server database, the port remains in the unauthorized state. However, if the client MAC address is added to the database, the switch can use MAC authentication bypass to re-authorize the port.
- If the port is in the authorized state, the port remains in this state until re-authorization occurs.
- You can configure a timeout period for hosts that are connected by MAC authentication bypass but are inactive. The range is 1 to 65535 seconds.

Maximum Number of Allowed Devices Per Port

This is the maximum number of devices allowed on an 802.1x-enabled port:

- In single-host mode, only one device is allowed on the access VLAN. If the port is also configured with a voice VLAN, an unlimited number of Cisco IP phones can send and receive traffic through the voice VLAN.
- In multidomain authentication (MDA) mode, one device is allowed for the access VLAN, and one IP phone is allowed for the voice VLAN.
- In multihost mode, only one 802.1x supplicant is allowed on the port, but an unlimited number of non-802.1x hosts are allowed on the access VLAN. An unlimited number of devices are allowed on the voice VLAN.

Configuring 802.1x Readiness Check

The 802.1x readiness check monitors 802.1x activity on all the switch ports and displays information about the devices connected to the ports that support 802.1x. You can use this feature to determine if the devices connected to the switch ports are 802.1x-capable.

The 802.1x readiness check is allowed on all ports that can be configured for 802.1x. The readiness check is not available on a port that is configured as **dot1x force-unauthorized**.

Follow these steps to enable the 802.1x readiness check on the switch:

Before you begin

Follow these guidelines to enable the readiness check on the switch:

- The readiness check is typically used before 802.1x is enabled on the switch.
- If you use the **dot1x test eapol-capable** privileged EXEC command without specifying an interface, all the ports on the switch stack are tested.
- When you configure the **dot1x test eapol-capable** command on an 802.1x-enabled port, and the link comes up, the port queries the connected client about its 802.1x capability. When the client responds with a notification packet, it is 802.1x-capable. A syslog message is generated if the client responds within the timeout period. If the client does not respond to the query, the client is not 802.1x-capable. No syslog message is generated.
- When you configure the **dot1x test eapol-capable** command on an 802.1x-enabled port, and the link comes up, the port queries the connected client about its 802.1x capability. When the client responds with a notification packet, it is 802.1x-capable. A syslog message is generated if the client responds within the timeout period. If the client does not respond to the query, the client is not 802.1x-capable. No syslog message is generated.
- The readiness check can be sent on a port that handles multiple hosts (for example, a PC that is connected to an IP phone). A syslog message is generated for each of the clients that respond to the readiness check within the timer period.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **dot1x test eapol-capable** [interface *interface-id*]
4. **dot1x test timeout** *timeout*
5. **end**
6. **show running-config**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	dot1x test eapol-capable [interface <i>interface-id</i>] Example: Device# dot1x test eapol-capable interface gigabitethernet1/0/13	Enables the 802.1x readiness check on the switch. (Optional) For <i>interface-id</i> specify the port on which to check for IEEE 802.1x readiness.

	Command or Action	Purpose
	<pre>DOT1X_PORT_EAPOL_CAPABLE:DOT1X: MAC 00-01-02-4b-f1-a3 on gigabitethernet1/0/13 is EAPOL capable</pre>	Note If you omit the optional interface keyword, all interfaces on the switch are tested.
Step 4	dot1x test timeout <i>timeout</i>	(Optional) Configures the timeout used to wait for EAPOL response. The range is from 1 to 65535 seconds. The default is 10 seconds.
Step 5	end Example: <pre>Device(config)# end</pre>	Returns to privileged EXEC mode.
Step 6	show running-config Example: <pre>Device# show running-config</pre>	Verifies your entries.
Step 7	copy running-config startup-config Example: <pre>Device# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Configuring Voice Aware 802.1x Security



Note To use voice aware IEEE 802.1x authentication, the switch must be running the LAN base image.

You use the voice aware 802.1x security feature on the switch to disable only the VLAN on which a security violation occurs, whether it is a data or voice VLAN. You can use this feature in IP phone deployments where a PC is connected to the IP phone. A security violation found on the data VLAN results in the shutdown of only the data VLAN. The traffic on the voice VLAN flows through the switch without interruption.

Follow these guidelines to configure voice aware 802.1x voice security on the switch:

- You enable voice aware 802.1x security by entering the **errdisable detect cause security-violation shutdown vlan** global configuration command. You disable voice aware 802.1x security by entering the **no** version of this command. This command applies to all 802.1x-configured ports in the switch.



Note If you do not include the **shutdown vlan** keywords, the entire port is shut down when it enters the error-disabled state.

- If you use the **errdisable recovery cause security-violation** global configuration command to configure error-disabled recovery, the port is automatically re-enabled. If error-disabled recovery is not configured for the port, you re-enable it by using the **shutdown** and **no shutdown** interface configuration commands.
- You can re-enable individual VLANs by using the **clear errdisable interface interface-id vlan [vlan-list]** privileged EXEC command. If you do not specify a range, all VLANs on the port are enabled.

Beginning in privileged EXEC mode, follow these steps to enable voice aware 802.1x security:

SUMMARY STEPS

1. **configure terminal**
2. **errdisable detect cause security-violation shutdown vlan**
3. **errdisable recovery cause security-violation**
4. **clear errdisable interface interface-id vlan [vlan-list]**
5. Enter the following:
 - **shutdown**
 - **no shutdown**
6. **end**
7. **show errdisable detect**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal	Enter global configuration mode.
Step 2	errdisable detect cause security-violation shutdown vlan	Shut down any VLAN on which a security violation error occurs. Note If the shutdown vlan keywords are not included, the entire port enters the error-disabled state and shuts down.
Step 3	errdisable recovery cause security-violation	Enter global configuration mode.
Step 4	clear errdisable interface interface-id vlan [vlan-list]	(Optional) Reenable individual VLANs that have been error disabled. <ul style="list-style-type: none"> • For interface-id specify the port on which to reenable individual VLANs. • (Optional) For vlan-list specify a list of VLANs to be re-enabled. If vlan-list is not specified, all VLANs are re-enabled.
Step 5	Enter the following: <ul style="list-style-type: none"> • shutdown • no shutdown 	(Optional) Re-enable an error-disabled VLAN, and clear all error-disable indications.
Step 6	end	Return to privileged EXEC mode.
Step 7	show errdisable detect	Verify your entries.

Example

This example shows how to configure the switch to shut down any VLAN on which a security violation error occurs:

```
Switch(config)# errdisable detect cause security-violation shutdown vlan
```

This example shows how to re-enable all VLANs that were error disabled on port Gigabit Ethernet 40/2.

```
Switch# clear errdisable interface gigabitethernet40/2
vlan
```

You can verify your settings by entering the **show errdisable detect** privileged EXEC command.

Configuring 802.1x Violation Modes

You can configure an 802.1x port so that it shuts down, generates a syslog error, or discards packets from a new device when:

- a device connects to an 802.1x-enabled port
- the maximum number of allowed about devices have been authenticated on the port

Beginning in privileged EXEC mode, follow these steps to configure the security violation actions on the switch:

SUMMARY STEPS

1. **configure terminal**
2. **aaa new-model**
3. **aaa authentication dot1x {default} method1**
4. **interface interface-id**
5. **switchport mode access**
6. **authentication violation {shutdown | restrict | protect | replace}**
7. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 2	aaa new-model Example: Device(config)# aaa new-model	Enables AAA.

	Command or Action	Purpose
Step 3	aaa authentication dot1x {default} method1 Example: <pre>Device(config)# aaa authentication dot1x default group radius</pre>	<p>Creates an 802.1x authentication method list.</p> <p>To create a default list that is used when a named list is <i>not</i> specified in the authentication command, use the default keyword followed by the method that is to be used in default situations. The default method list is automatically applied to all ports.</p> <p>For <i>method1</i>, enter the group radius keywords to use the list of all RADIUS servers for authentication.</p>
Step 4	interface interface-id Example: <pre>Device(config)# interface gigabitethernet1/0/4</pre>	<p>Specifies the port connected to the client that is to be enabled for IEEE 802.1x authentication, and enter interface configuration mode.</p>
Step 5	switchport mode access Example: <pre>Device(config-if)# switchport mode access</pre>	<p>Sets the port to access mode.</p>
Step 6	authentication violation {shutdown restrict protect replace} Example: <pre>Device(config-if)# authentication violation restrict</pre>	<p>Configures the violation mode. The keywords have these meanings:</p> <ul style="list-style-type: none"> • shutdown—Error disable the port. • restrict—Generate a syslog error. • protect—Drop packets from any new device that sends traffic to the port. • replace—Removes the current session and authenticates with the new host.
Step 7	end Example: <pre>Device(config-if)# end</pre>	<p>Returns to privileged EXEC mode.</p>

Configuring 802.1x Authentication

To allow per-user ACLs or VLAN assignment, you must enable AAA authorization to configure the switch for all network-related service requests.

This is the 802.1x AAA process:

Before you begin

To configure 802.1x port-based authentication, you must enable authentication, authorization, and accounting (AAA) and specify the authentication method list. A method list describes the sequence and authentication method to be queried to authenticate a user.

SUMMARY STEPS

1. A user connects to a port on the switch.
2. Authentication is performed.
3. VLAN assignment is enabled, as appropriate, based on the RADIUS server configuration.
4. The switch sends a start message to an accounting server.
5. Re-authentication is performed, as necessary.
6. The switch sends an interim accounting update to the accounting server that is based on the result of re-authentication.
7. The user disconnects from the port.
8. The switch sends a stop message to the accounting server.

DETAILED STEPS

	Command or Action	Purpose
Step 1	A user connects to a port on the switch.	
Step 2	Authentication is performed.	
Step 3	VLAN assignment is enabled, as appropriate, based on the RADIUS server configuration.	
Step 4	The switch sends a start message to an accounting server.	
Step 5	Re-authentication is performed, as necessary.	
Step 6	The switch sends an interim accounting update to the accounting server that is based on the result of re-authentication.	
Step 7	The user disconnects from the port.	
Step 8	The switch sends a stop message to the accounting server.	

Configuring 802.1x Port-Based Authentication

Beginning in privileged EXEC mode, follow these steps to configure 802.1x port-based authentication:

SUMMARY STEPS

1. **configure terminal**
2. **aaa new-model**
3. **aaa authentication dot1x {default} *method1***
4. **dot1x system-auth-control**
5. **aaa authorization network {default} group radius**
6. **radius-server host *ip-address***

7. `radius-server key string`
8. `interface interface-id`
9. `switchport mode access`
10. `authentication port-control auto`
11. `dot1x pae authenticator`
12. `end`

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode.
Step 2	aaa new-model Example: Device(config)# <code>aaa new-model</code>	Enables AAA.
Step 3	aaa authentication dot1x {default} method1 Example: Device(config)# <code>aaa authentication dot1x default group radius</code>	<p>Creates an 802.1x authentication method list.</p> <p>To create a default list that is used when a named list is <i>not</i> specified in the authentication command, use the default keyword followed by the method that is to be used in default situations. The default method list is automatically applied to all ports.</p> <p>For <i>method1</i>, enter the group radius keywords to use the list of all RADIUS servers for authentication.</p> <p>Note Though other keywords are visible in the command-line help string, only the group radius keywords are supported.</p>
Step 4	dot1x system-auth-control Example: Device(config)# <code>dot1x system-auth-control</code>	Enables 802.1x authentication globally on the switch.
Step 5	aaa authorization network {default} group radius Example: Device(config)# <code>aaa authorization network default group radius</code>	(Optional) Configures the switch to use user-RADIUS authorization for all network-related service requests, such as per-user ACLs or VLAN assignment.

	Command or Action	Purpose
Step 6	radius-server host <i>ip-address</i> Example: Device(config)# radius-server host 124.2.2.12	(Optional) Specifies the IP address of the RADIUS server.
Step 7	radius-server key <i>string</i> Example: Device(config)# radius-server key abc1234	(Optional) Specifies the authentication and encryption key used between the switch and the RADIUS daemon running on the RADIUS server.
Step 8	interface <i>interface-id</i> Example: Device(config)# interface gigabitethernet1/0/2	Specifies the port connected to the client that is to be enabled for IEEE 802.1x authentication, and enter interface configuration mode.
Step 9	switchport mode access Example: Device(config-if)# switchport mode access	(Optional) Sets the port to access mode only if you configured the RADIUS server in Step 6 and Step 7.
Step 10	authentication port-control auto Example: Device(config-if)# authentication port-control auto	Enables 802.1x authentication on the port.
Step 11	dot1x pae authenticator Example: Device(config-if)# dot1x pae authenticator	Sets the interface Port Access Entity to act only as an authenticator and ignore messages meant for a supplicant.
Step 12	end Example: Device(config-if)# end	Returns to privileged EXEC mode.

Configuring the Switch-to-RADIUS-Server Communication

You can globally configure the timeout, retransmission, and encryption key values for all RADIUS servers by using the **radius-server host** global configuration command. If you want to configure these options on a

per-server basis, use the **radius-server timeout**, the **radius-server retransmit**, and the **radius-server key** global configuration commands.

You also need to configure some settings on the RADIUS server. These settings include the IP address of the switch and the key string to be shared by both the server and the switch. For more information, see the RADIUS server documentation.

Follow these steps to configure the RADIUS server parameters on the switch. This procedure is required.

Before you begin

You must enable authentication, authorization, and accounting (AAA) and specify the authentication method list. A method list describes the sequence and authentication method to be queried to authenticate a user.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **radius-server host** *{hostname | ip-address}* **auth-port** *port-number* **key** *string*
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	radius-server host <i>{hostname ip-address}</i> auth-port <i>port-number</i> key <i>string</i> Example: Device(config)# radius-server host 125.5.5.43 auth-port 1645 key rad123	Configures the RADIUS server parameters. For <i>hostname ip-address</i> , specify the server name or IP address of the remote RADIUS server. For auth-port <i>port-number</i> , specify the UDP destination port for authentication requests. The default is 1645. The range is 0 to 65536. For key <i>string</i> , specify the authentication and encryption key used between the switch and the RADIUS daemon running on the RADIUS server. The key is a text string that must match the encryption key used on the RADIUS server.

	Command or Action	Purpose
		<p>Note Always configure the key as the last item in the radius-server host command syntax because leading spaces are ignored, but spaces within and at the end of the key are used. If you use spaces in the key, do not enclose the key in quotation marks unless the quotation marks are part of the key. This key must match the encryption used on the RADIUS daemon.</p> <p>If you want to use multiple RADIUS servers, re-enter this command.</p>
Step 4	<p>end</p> <p>Example:</p> <pre>Device(config)# end</pre>	Returns to privileged EXEC mode.

Configuring the Host Mode

Beginning in privileged EXEC mode, follow these steps to allow multiple hosts (clients) on an IEEE 802.1x-authorized port that has the **authentication port-control** interface configuration command set to **auto**. Use the **multi-domain** keyword to configure and enable multidomain authentication (MDA), which allows both a host and a voice device, such as an IP phone (Cisco or non-Cisco), on the same switch port. This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **authentication host-mode** [**multi-auth** | **multi-domain** | **multi-host** | **single-host**]
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p>configure terminal</p> <p>Example:</p> <pre>Device# configure terminal</pre>	Enters global configuration mode.
Step 2	<p>interface <i>interface-id</i></p> <p>Example:</p> <pre>Device(config)# interface gigabitethernet2/0/1</pre>	Specifies the port to which multiple hosts are indirectly attached, and enter interface configuration mode.

	Command or Action	Purpose
Step 3	<p>authentication host-mode [multi-auth multi-domain multi-host single-host]</p> <p>Example:</p> <pre>Device(config-if)# authentication host-mode multi-host</pre>	<p>Allows multiple hosts (clients) on an 802.1x-authorized port.</p> <p>The keywords have these meanings:</p> <ul style="list-style-type: none"> • multi-auth—Allow multiple authenticated clients on both the voice VLAN and data VLAN. <p>Note The multi-auth keyword is only available with the authentication host-mode command.</p> <ul style="list-style-type: none"> • multi-host—Allow multiple hosts on an 802.1x-authorized port after a single host has been authenticated. • multi-domain—Allow both a host and a voice device, such as an IP phone (Cisco or non-Cisco), to be authenticated on an IEEE 802.1x-authorized port. <p>Note You must configure the voice VLAN for the IP phone when the host mode is set to multi-domain.</p> <p>Make sure that the authentication port-control interface configuration command is set to auto for the specified interface.</p>
Step 4	<p>end</p> <p>Example:</p> <pre>Device(config-if)# end</pre>	<p>Returns to privileged EXEC mode.</p>

Configuring Periodic Re-Authentication

You can enable periodic 802.1x client re-authentication and specify how often it occurs. If you do not specify a time period before enabling re-authentication, the number of seconds between attempts is 3600.

Beginning in privileged EXEC mode, follow these steps to enable periodic re-authentication of the client and to configure the number of seconds between re-authentication attempts. This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **authentication periodic**
4. **authentication timer** {{{[inactivity | reauthenticate | restart | unauthorized]}} {value}}
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode.
Step 2	interface <i>interface-id</i> Example: Device(config)# <code>interface gigabitethernet2/0/1</code>	Specifies the port to be configured, and enter interface configuration mode.
Step 3	authentication periodic Example: Device(config-if)# <code>authentication periodic</code>	Enables periodic re-authentication of the client, which is disabled by default. Note The default value is 3600 seconds. To change the value of the reauthentication timer or to have the switch use a RADIUS-provided session timeout, enter the authentication timer reauthenticate command.
Step 4	authentication timer {[inactivity reauthenticate restart unauthorized]} {value}} Example: Device(config-if)# <code>authentication timer reauthenticate 180</code>	Sets the number of seconds between re-authentication attempts. The authentication timer keywords have these meanings: <ul style="list-style-type: none"> • inactivity—Interval in seconds after which if there is no activity from the client then it is unauthorized • reauthenticate—Time in seconds after which an automatic re-authentication attempt is initiated • restart <i>value</i>—Interval in seconds after which an attempt is made to authenticate an unauthorized port • unauthorized <i>value</i>—Interval in seconds after which an unauthorized session will get deleted This command affects the behavior of the switch only if periodic re-authentication is enabled.
Step 5	end Example: Device(config-if)# <code>end</code>	Returns to privileged EXEC mode.

Changing the Quiet Period

When the switch cannot authenticate the client, the switch remains idle for a set period of time and then tries again. The **authentication timer restart** interface configuration command controls the idle period. A failed authentication of the client might occur because the client provided an invalid password. You can provide a faster response time to the user by entering a number smaller than the default.

Beginning in privileged EXEC mode, follow these steps to change the quiet period. This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **authentication timer restart** *seconds*
4. **end**
5. **show authentication sessions interface** *interface-id*
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 2	interface <i>interface-id</i> Example: Device(config)# interface gigabitethernet2/0/1	Specifies the port to be configured, and enter interface configuration mode.
Step 3	authentication timer restart <i>seconds</i> Example: Device(config-if)# authentication timer restart 30	Sets the number of seconds that the switch remains in the quiet state following a failed authentication exchange with the client. The range is 1 to 65535 seconds; the default is 60.
Step 4	end Example: Device(config-if)# end	Returns to privileged EXEC mode.
Step 5	show authentication sessions interface <i>interface-id</i> Example: Device# show authentication sessions interface	Verifies your entries.

	Command or Action	Purpose
	<code>gigabitethernet2/0/1</code>	
Step 6	copy running-config startup-config Example: Device# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Changing the Switch-to-Client Retransmission Time

The client responds to the EAP-request/identity frame from the switch with an EAP-response/identity frame. If the switch does not receive this response, it waits a set period of time (known as the retransmission time) and then resends the frame.



Note You should change the default value of this command only to adjust for unusual circumstances such as unreliable links or specific behavioral problems with certain clients and authentication servers.

Beginning in privileged EXEC mode, follow these steps to change the amount of time that the switch waits for client notification. This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **authentication timer reauthenticate** *seconds*
4. **end**
5. **show authentication sessions interface** *interface-id*
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode.
Step 2	interface <i>interface-id</i> Example: Device(config)# <code>interface gigabitethernet2/0/1</code>	Specifies the port to be configured, and enter interface configuration mode.

	Command or Action	Purpose
Step 3	authentication timer reauthenticate <i>seconds</i> Example: <pre>Device(config-if)# authentication timer reauthenticate 60</pre>	Sets the number of seconds that the switch waits for a response to an EAP-request/identity frame from the client before resending the request. The range is 1 to 65535 seconds; the default is 5.
Step 4	end Example: <pre>Device(config-if)# end</pre>	Returns to privileged EXEC mode.
Step 5	show authentication sessions interface <i>interface-id</i> Example: <pre>Device# show authentication sessions interface gigabitethernet2/0/1</pre>	Verifies your entries.
Step 6	copy running-config startup-config Example: <pre>Device# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Setting the Switch-to-Client Frame-Retransmission Number

In addition to changing the switch-to-client retransmission time, you can change the number of times that the switch sends an EAP-request/identity frame (assuming no response is received) to the client before restarting the authentication process.



Note You should change the default value of this command only to adjust for unusual circumstances such as unreliable links or specific behavioral problems with certain clients and authentication servers.

Beginning in privileged EXEC mode, follow these steps to set the switch-to-client frame-retransmission number. This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **dot1x max-reauth-req** *count*
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode.
Step 2	interface <i>interface-id</i> Example: Device(config)# <code>interface gigabitethernet2/0/1</code>	Specifies the port to be configured, and enter interface configuration mode.
Step 3	dot1x max-reauth-req <i>count</i> Example: Device(config-if)# <code>dot1x max-reauth-req 5</code>	Sets the number of times that the switch sends an EAP-request/identity frame to the client before restarting the authentication process. The range is 1 to 10; the default is 2.
Step 4	end Example: Device(config-if)# <code>end</code>	Returns to privileged EXEC mode.

Setting the Re-Authentication Number

You can also change the number of times that the switch restarts the authentication process before the port changes to the unauthorized state.



Note You should change the default value of this command only to adjust for unusual circumstances such as unreliable links or specific behavioral problems with certain clients and authentication servers.

Beginning in privileged EXEC mode, follow these steps to set the re-authentication number. This procedure is optional.

SUMMARY STEPS

1. `configure terminal`
2. `interface interface-id`
3. `switchport mode access`
4. `dot1x max-req count`
5. `end`

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode.
Step 2	interface <i>interface-id</i> Example: Device# <code>interface gigabitethernet2/0/1</code>	Specifies the port to be configured, and enter interface configuration mode.
Step 3	switchport mode access Example: Device(config-if)# <code>switchport mode access</code>	Sets the port to access mode only if you previously configured the RADIUS server.
Step 4	dot1x max-req <i>count</i> Example: Device(config-if)# <code>dot1x max-req 4</code>	Sets the number of times that the switch restarts the authentication process before the port changes to the unauthorized state. The range is 0 to 10; the default is 2.
Step 5	end Example: Device(config-if)# <code>end</code>	Returns to privileged EXEC mode.

Enabling MAC Move

MAC move allows an authenticated host to move from one port on the switch to another.

Beginning in privileged EXEC mode, follow these steps to globally enable MAC move on the switch. This procedure is optional.

SUMMARY STEPS

1. `configure terminal`
2. `authentication mac-move permit`
3. `end`
4. `show running-config`
5. `copy running-config startup-config`

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode.
Step 2	authentication mac-move permit Example: Device(config)# <code>authentication mac-move permit</code>	Enables MAC move on the switch. Default is deny. In Session Aware Networking mode, the default CLI is access-session mac-move deny . To enable Mac Move in Session Aware Networking, use the no access-session mac-move global configuration command. In legacy mode (IBNS 1.0), default value for mac-move is deny and in C3PL mode (IBNS 2.0) default value is permit .
Step 3	end Example: Device(config)# <code>end</code>	Returns to privileged EXEC mode.
Step 4	show running-config Example: Device# <code>show running-config</code>	Verifies your entries.
Step 5	copy running-config startup-config Example: Device# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Disabling MAC Move

To disable MAC move from a secure port to an unsecured port on a switch, beginning in privileged EXEC mode, follow these steps. This procedure is optional.

SUMMARY STEPS

1. `configure terminal`
2. `authentication mac-move deny-uncontrolled`
3. `end`
4. `show running-config`
5. `copy running-config startup-config`

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode.
Step 2	authentication mac-move deny-uncontrolled Example: Device(config)# <code>authentication mac-move deny-uncontrolled</code>	Disables MAC move on the switch.
Step 3	end Example: Device(config)# <code>end</code>	Returns to privileged EXEC mode.
Step 4	show running-config Example: Device# <code>show running-config</code>	Verifies your entries.
Step 5	copy running-config startup-config Example: Device# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Enabling MAC Replace

MAC replace allows a host to replace an authenticated host on a port.

Beginning in privileged EXEC mode, follow these steps to enable MAC replace on an interface. This procedure is optional.

SUMMARY STEPS

1. `configure terminal`
2. `interface interface-id`
3. `authentication violation {protect | replace | restrict | shutdown}`
4. `end`
5. `show running-config`
6. `copy running-config startup-config`

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode.
Step 2	interface <i>interface-id</i> Example: Device(config)# <code>interface gigabitethernet2/0/2</code>	Specifies the port to be configured, and enter interface configuration mode.
Step 3	authentication violation {protect replace restrict shutdown} Example: Device(config-if)# <code>authentication violation replace</code>	Use the replace keyword to enable MAC replace on the interface. The port removes the current session and initiates authentication with the new host. The other keywords have these effects: <ul style="list-style-type: none"> • protect: the port drops packets with unexpected MAC addresses without generating a system message. • restrict: violating packets are dropped by the CPU and a system message is generated. • shutdown: the port is error disabled when it receives an unexpected MAC address.
Step 4	end Example: Device(config-if)# <code>end</code>	Returns to privileged EXEC mode.
Step 5	show running-config Example: Device# <code>show running-config</code>	Verifies your entries.
Step 6	copy running-config startup-config Example: Device# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Configuring 802.1x Accounting

Enabling AAA system accounting with 802.1x accounting allows system reload events to be sent to the accounting RADIUS server for logging. The server can then infer that all active 802.1x sessions are closed.

Because RADIUS uses the unreliable UDP transport protocol, accounting messages might be lost due to poor network conditions. If the switch does not receive the accounting response message from the RADIUS server after a configurable number of retransmissions of an accounting request, this system message appears:

```
Accounting message %s for session %s failed to receive Accounting Response.
```

When the stop message is not sent successfully, this message appears:

```
00:09:55: %RADIUS-4-RADIUS_DEAD: RADIUS server 172.20.246.201:1645,1646 is not responding.
```



Note You must configure the RADIUS server to perform accounting tasks, such as logging start, stop, and interim-update messages and time stamps. To turn on these functions, enable logging of “Update/Watchdog packets from this AAA client” in your RADIUS server Network Configuration tab. Next, enable “CVS RADIUS Accounting” in your RADIUS server System Configuration tab.

Beginning in privileged EXEC mode, follow these steps to configure 802.1x accounting after AAA is enabled on your switch. This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **interface *interface-id***
3. **aaa accounting dot1x default start-stop group radius**
4. **aaa accounting system default start-stop group radius**
5. **end**
6. **show running-config**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 2	interface <i>interface-id</i> Example: Device(config)# interface gigabitethernet1/0/3	Specifies the port to be configured, and enter interface configuration mode.

	Command or Action	Purpose
Step 3	aaa accounting dot1x default start-stop group radius Example: <pre>Device(config-if)# aaa accounting dot1x default start-stop group radius</pre>	Enables 802.1x accounting using the list of all RADIUS servers.
Step 4	aaa accounting system default start-stop group radius Example: <pre>Device(config-if)# aaa accounting system default start-stop group radius</pre>	(Optional) Enables system accounting (using the list of all RADIUS servers) and generates system accounting reload event messages when the switch reloads.
Step 5	end Example: <pre>Device(config-if)# end</pre>	Returns to privileged EXEC mode.
Step 6	show running-config Example: <pre>Device# show running-config</pre>	Verifies your entries.
Step 7	copy running-config startup-config Example: <pre>Device# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Configuring a Guest VLAN

When you configure a guest VLAN, clients that are not 802.1x-capable are put into the guest VLAN when the server does not receive a response to its EAP request/identity frame. Clients that are 802.1x-capable but that fail authentication are not granted network access. The switch supports guest VLANs in single-host or multiple-hosts mode.

Beginning in privileged EXEC mode, follow these steps to configure a guest VLAN. This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. Use one of the following:
 - **switchport mode access**
 - **switchport mode private-vlan host**

4. **authentication event no-response action authorize vlan** *vlan-id*
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 2	interface <i>interface-id</i> Example: Device(config)# interface gigabitethernet 2/0/2	Specifies the port to be configured, and enter interface configuration mode.
Step 3	Use one of the following: <ul style="list-style-type: none"> • switchport mode access • switchport mode private-vlan host Example: Device(config-if)# switchport mode private-vlan host	<ul style="list-style-type: none"> • Sets the port to access mode. • Configures the Layer 2 port as a private-VLAN host port.
Step 4	authentication event no-response action authorize vlan <i>vlan-id</i> Example: Device(config-if)# authentication event no-response action authorize vlan 2	Specifies an active VLAN as an 802.1x guest VLAN. The range is 1 to 4094. You can configure any active VLAN except an internal VLAN (routed port), an RSPAN VLAN or a voice VLAN as an 802.1x guest VLAN.
Step 5	end Example: Device(config-if)# end	Returns to privileged EXEC mode.

Configuring a Restricted VLAN

When you configure a restricted VLAN on a switch stack or a switch, clients that are IEEE 802.1x-compliant are moved into the restricted VLAN when the authentication server does not receive a valid username and password. The switch supports restricted VLANs only in single-host mode.

Beginning in privileged EXEC mode, follow these steps to configure a restricted VLAN. This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. Use one of the following:
 - **switchport mode access**
 - **switchport mode private-vlan host**
4. **authentication port-control auto**
5. **authentication event fail action authorize vlan** *vlan-id*
6. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 2	interface <i>interface-id</i> Example: Device(config)# interface gigabitethernet 2/0/2	Specifies the port to be configured, and enter interface configuration mode.
Step 3	Use one of the following: <ul style="list-style-type: none"> • switchport mode access • switchport mode private-vlan host Example: Device(config-if)# switchport mode access	<ul style="list-style-type: none"> • Sets the port to access mode. • Configures the Layer 2 port as a private-VLAN host port.
Step 4	authentication port-control auto Example: Device(config-if)# authentication port-control auto	Enables 802.1x authentication on the port.
Step 5	authentication event fail action authorize vlan <i>vlan-id</i> Example: Device(config-if)# authentication event fail action authorize vlan 2	Specifies an active VLAN as an 802.1x restricted VLAN. The range is 1 to 4094. You can configure any active VLAN except an internal VLAN (routed port), an RSPAN VLAN or a voice VLAN as an 802.1x restricted VLAN.

	Command or Action	Purpose
Step 6	end Example: Device(config-if)# end	Returns to privileged EXEC mode.

Configuring Number of Authentication Attempts on a Restricted VLAN

You can configure the maximum number of authentication attempts allowed before a user is assigned to the restricted VLAN by using the **authentication event retry** *retry count* interface configuration command. The range of allowable authentication attempts is 1 to 3. The default is 3 attempts.

Beginning in privileged EXEC mode, follow these steps to configure the maximum number of allowed authentication attempts. This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. Use one of the following:
 - **switchport mode access**
 - **switchport mode private-vlan host**
4. **authentication port-control auto**
5. **authentication event fail action authorize vlan** *vlan-id*
6. **authentication event retry** *retry count*
7. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 2	interface <i>interface-id</i> Example: Device(config)# interface gigabitethernet 2/0/3	Specifies the port to be configured, and enter interface configuration mode.
Step 3	Use one of the following: <ul style="list-style-type: none"> • switchport mode access • switchport mode private-vlan host Example:	<ul style="list-style-type: none"> • Sets the port to access mode. • Configures the Layer 2 port as a private-VLAN host port.

	Command or Action	Purpose
	OR Device (config-if) # switchport mode access	
Step 4	authentication port-control auto Example: Device (config-if) # authentication port-control auto	Enables 802.1x authentication on the port.
Step 5	authentication event fail action authorize vlan <i>vlan-id</i> Example: Device (config-if) # authentication event fail action authorize vlan 8	Specifies an active VLAN as an 802.1x restricted VLAN. The range is 1 to 4094. You can configure any active VLAN except an internal VLAN (routed port), an RSPAN VLAN or a voice VLAN as an 802.1x restricted VLAN.
Step 6	authentication event retry <i>retry count</i> Example: Device (config-if) # authentication event retry 2	Specifies a number of authentication attempts to allow before a port moves to the restricted VLAN. The range is 1 to 3, and the default is 3.
Step 7	end Example: Device (config-if) # end	Returns to privileged EXEC mode.

Configuring 802.1x Inaccessible Authentication Bypass with Critical Voice VLAN

Beginning in privileged EXEC mode, follow these steps to configure critical voice VLAN on a port and enable the inaccessible authentication bypass feature.

SUMMARY STEPS

1. **configure terminal**
2. **aaa new-model**
3. **radius-server dead-criteria {time *seconds* } [*tries number*]**
4. **radius-server deadtime *minutes***
5. **radius-server host ip-address *address* [*acct-port udp-port*] [*auth-port udp-port*] [*testusername name*] [*idle-time time*] [*ignore-acct-port*] [*ignore auth-port*] [*key string*]**
6. **dot1x critical {eapol | recovery delay *milliseconds*}**
7. **interface *interface-id***

8. **authentication event server dead action** {authorize | reinitialize} **vlan** *vlan-id*
9. **switchport voice vlan** *vlan-id*
10. **authentication event server dead action authorize voice**
11. **show authentication interface** *interface-id*
12. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 2	aaa new-model Example: Device(config)# aaa new-model	Enables AAA.
Step 3	radius-server dead-criteria {time <i>seconds</i> } [<i>tries number</i>] Example: Device(config)# radius-server dead-criteria time 20 tries 10	Sets the conditions that determine when a RADIUS server is considered un-available or down (dead). <ul style="list-style-type: none"> • time— 1 to 120 seconds. The switch dynamically determines a default <i>seconds</i> value between 10 and 60. • number—1 to 100 tries. The switch dynamically determines a default <i>triesnumber</i> between 10 and 100.
Step 4	radius-server deadtime <i>minutes</i> Example: Device(config)# radius-server deadtime 60	(Optional) Sets the number of minutes during which a RADIUS server is not sent requests. The range is from 0 to 1440 minutes (24 hours). The default is 0 minutes.
Step 5	radius-server host ip-address <i>address</i> [<i>acct-port udp-port</i>] [<i>auth-port udp-port</i>] [<i>testusername name</i>] [<i>idle-time time</i>] [<i>ignore-acct-port</i>] [<i>ignore auth-port</i>] [<i>key string</i>] Example: Device(config)# radius-server host 10.0.0.10 acct-port 1550 auth-port 1560 test username user1 idle-time 30 key abc1234	(Optional) Configure the RADIUS server parameters by using these keywords: <ul style="list-style-type: none"> • acct-port <i>udp-port</i>—Specify the UDP port for the RADIUS accounting server. The range for the UDP port number is from 0 to 65536. The default is 1646. • auth-port <i>udp-port</i>—Specify the UDP port for the RADIUS authentication server. The range for the UDP port number is from 0 to 65536. The default is 1645.

	Command or Action	Purpose
		<p>Note You should configure the UDP port for the RADIUS accounting server and the UDP port for the RADIUS authentication server to nondefault values.</p> <ul style="list-style-type: none"> • test username <i>name</i>—Enable automated testing of the RADIUS server status, and specify the username to be used. • idle-time <i>time</i>—Set the interval of time in minutes after which the switch sends test packets to the server. The range is from 1 to 35791 minutes. The default is 60 minutes (1 hour). • ignore-acct-port—Disable testing on the RADIUS-server accounting port. • ignore-auth-port—Disable testing on the RADIUS-server authentication port. • For key <i>string</i>, specify the authentication and encryption key used between the switch and the RADIUS daemon running on the RADIUS server. The key is a text string that must match the encryption key used on the RADIUS server. <p>Note Always configure the key as the last item in the radius-server host command syntax because leading spaces are ignored, but spaces within and at the end of the key are used. If you use spaces in the key, do not enclose the key in quotation marks unless the quotation marks are part of the key. This key must match the encryption used on the RADIUS daemon.</p> <p>You can also configure the authentication and encryption key by using the radius-server key {<i>0string</i> <i>7string</i> <i>string</i>} global configuration command.</p>
Step 6	<p>dot1x critical {<i>eapol</i> recovery delay <i>milliseconds</i>}</p> <p>Example:</p> <pre>Device(config)# dot1x critical eapol (config)# dot1x critical recovery delay 2000</pre>	<p>(Optional) Configure the parameters for inaccessible authentication bypass:</p> <ul style="list-style-type: none"> • eapol—Specify that the switch sends an EAPOL-Success message when the switch successfully authenticates the critical port. • recovery delay <i>milliseconds</i>—Set the recovery delay period during which the switch waits to re-initialize a critical port when a RADIUS server that was unavailable becomes available. The range is from 1 to 10000 milliseconds. The default is 1000

	Command or Action	Purpose
		milliseconds (a port can be re-initialized every second).
Step 7	interface <i>interface-id</i> Example: Device(config)# interface gigabitethernet 1/0/1	Specify the port to be configured, and enter interface configuration mode.
Step 8	authentication event server dead action { authorize reinitialize } vlan <i>vlan-id</i>] Example: Device(config-if)# authentication event server dead action reinitialicze vlan 20	Use these keywords to move hosts on the port if the RADIUS server is unreachable: <ul style="list-style-type: none"> • authorize—Move any new hosts trying to authenticate to the user-specified critical VLAN. • reinitialize—Move all authorized hosts on the port to the user-specified critical VLAN.
Step 9	switchport voice vlan <i>vlan-id</i> Example: Device(config-if)# switchport voice vlan	Specifies the voice VLAN for the port. The voice VLAN cannot be the same as the critical data VLAN configured in Step 6.
Step 10	authentication event server dead action authorize voice Example: Device(config-if)# authentication event server dead action authorize voice	Configures critical voice VLAN to move data traffic on the port to the voice VLAN if the RADIUS server is unreachable.
Step 11	show authentication interface <i>interface-id</i> Example: Device(config-if)# do show authentication interface gigabit 1/0/1	(Optional) Verify your entries.
Step 12	copy running-config startup-config Example: Device(config-if)# do copy running-config startup-config	(Optional) Verify your entries.

Example

To return to the RADIUS server default settings, use the **no radius-server dead-criteria**, the **no radius-server deadtime**, and the **no radius-server host** global configuration commands. To disable

inaccessible authentication bypass, use the **no authentication event server dead action** interface configuration command. To disable critical voice VLAN, use the **no authentication event server dead action authorize voice** interface configuration command.

Example of Configuring Inaccessible Authentication Bypass

This example shows how to configure the inaccessible authentication bypass feature:

```
Device(config)# radius-server dead-criteria time 30 tries 20
Device(config)# radius-server deadtime 60
Device(config)# radius-server host 10.0.0.10 acct-port 1550 auth-port 1560 test username
user1 idle-time 30 key abc1234
Device(config)# dot1x critical eapol
Device(config)# dot1x critical recovery delay 2000
Device(config)# interface gigabitethernet 1/0/1
Device(config-if)# dot1x critical
Device(config-if)# dot1x critical recovery action reinitialize
Device(config-if)# dot1x critical vlan 20
Device(config-if)# end
```

Configuring 802.1x Authentication with WoL

Beginning in privileged EXEC mode, follow these steps to enable 802.1x authentication with WoL. This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **authentication control-direction** {both | in}
4. **end**
5. **show authentication sessions interface** *interface-id*
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 2	interface <i>interface-id</i> Example: Device(config)# interface gigabitethernet2/0/3	Specifies the port to be configured, and enter interface configuration mode.

	Command or Action	Purpose
Step 3	authentication control-direction {both in} Example: <pre>Device(config-if)# authentication control-direction both</pre>	Enables 802.1x authentication with WoL on the port, and use these keywords to configure the port as bidirectional or unidirectional. <ul style="list-style-type: none"> • both—Sets the port as bidirectional. The port cannot receive packets from or send packets to the host. By default, the port is bidirectional. • in—Sets the port as unidirectional. The port can send packets to the host but cannot receive packets from the host.
Step 4	end Example: <pre>Device(config-if)# end</pre>	Returns to privileged EXEC mode.
Step 5	show authentication sessions interface <i>interface-id</i> Example: <pre>Device# show authentication sessions interface gigabitethernet2/0/3</pre>	Verifies your entries.
Step 6	copy running-config startup-config Example: <pre>Device# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Configuring MAC Authentication Bypass

Beginning in privileged EXEC mode, follow these steps to enable MAC authentication bypass. This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **interface *interface-id***
3. **authentication port-control auto**
4. **mab [eap]**
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode.
Step 2	interface interface-id Example: Device(config)# <code>interface gigabitethernet 2/0/1</code>	Specifies the port to be configured, and enter interface configuration mode.
Step 3	authentication port-control auto Example: Device(config-if)# <code>authentication port-control auto</code>	Enables 802.1x authentication on the port.
Step 4	mab [eap] Example: Device(config-if)# <code>mab</code>	Enables MAC authentication bypass. (Optional) Use the eap keyword to configure the switch to use EAP for authorization.
Step 5	end Example: Device(config-if)# <code>end</code>	Returns to privileged EXEC mode.

Formatting a MAC Authentication Bypass Username and Password

Use the optional **mab request format** command to format the MAB username and password in a style accepted by the authentication server. The username and password are usually the MAC address of the client. Some authentication server configurations require the password to be different from the username.

Beginning in privileged EXEC mode, follow these steps to format MAC authentication bypass username and passwords.

SUMMARY STEPS

1. `configure terminal`
2. `mab request format attribute 1 groupsize {1 | 2 | 4 | 12} [separator {- | : | .}] {lowercase | uppercase}`
3. `mab request format attribute2 {0 | 7} text`
4. `end`

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>Device# configure terminal</pre>	Enters global configuration mode.
Step 2	mab request format attribute 1 groupsize {1 2 4 12} [separator {- : .} {lowercase uppercase}] Example: <pre>Device(config)# mab request format attribute 1 groupsize 12</pre>	<p>Specifies the format of the MAC address in the User-Name attribute of MAB-generated Access-Request packets.</p> <p>1—Sets the username format of the 12 hex digits of the MAC address.</p> <p>group size—The number of hex nibbles to concatenate before insertion of a separator. A valid groupsize must be either 1, 2, 4, or 12.</p> <p>separator—The character that separates the hex nibbles according to group size. A valid separator must be either a hyphen, colon, or period. No separator is used for a group size of 12.</p> <p>{lowercase uppercase}—Specifies if nonnumeric hex nibbles should be in lowercase or uppercase.</p>
Step 3	mab request format attribute2 {0 7} text Example: <pre>Device(config)# mab request format attribute 2 7 A02f44E18B12</pre>	<p>2—Specifies a custom (nondefault) value for the User-Password attribute in MAB-generated Access-Request packets.</p> <p>0—Specifies a cleartext password to follow.</p> <p>7—Specifies an encrypted password to follow.</p> <p>text—Specifies the password to be used in the User-Password attribute.</p> <p>Note When you send configuration information in e-mail, remove type 7 password information. The show tech-support command removes this information from its output by default.</p>
Step 4	end Example: <pre>Device(config)# end</pre>	Returns to privileged EXEC mode.

Configuring 802.1x User Distribution

Beginning in privileged EXEC mode, follow these steps to configure a VLAN group and to map a VLAN to it:

SUMMARY STEPS

1. **configure terminal**
2. **vlan group** *vlan-group-name* **vlan-list** *vlan-list*
3. **end**
4. **no vlan group** *vlan-group-name* **vlan-list** *vlan-list*

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 2	vlan group <i>vlan-group-name</i> vlan-list <i>vlan-list</i> Example: Device(config)# vlan group eng-dept vlan-list 10	Configures a VLAN group, and maps a single VLAN or a range of VLANs to it.
Step 3	end Example: Device(config)# end	Returns to privileged EXEC mode.
Step 4	no vlan group <i>vlan-group-name</i> vlan-list <i>vlan-list</i> Example: Device(config)# no vlan group eng-dept vlan-list 10	Clears the VLAN group configuration or elements of the VLAN group configuration.

Example of Configuring VLAN Groups

This example shows how to configure the VLAN groups, to map the VLANs to the groups, to and verify the VLAN group configurations and mapping to the specified VLANs:

```
Device(config)# vlan group eng-dept vlan-list 10

Device(config)# show vlan group group-name eng-dept
Group Name                Vlans Mapped
-----                -
eng-dept                    10

Device(config)# show dot1x vlan-group all
Group Name                Vlans Mapped
-----                -
eng-dept                    10
```

```
hr-dept                20
```

This example shows how to add a VLAN to an existing VLAN group and to verify that the VLAN was added:

```
Device(config)# vlan group eng-dept vlan-list 30
Device(config)# show vlan group eng-dept
Group Name          Vlans Mapped
-----
eng-dept            10,30
```

This example shows how to remove a VLAN from a VLAN group:

```
Device# no vlan group eng-dept vlan-list 10
```

This example shows that when all the VLANs are cleared from a VLAN group, the VLAN group is cleared:

```
Device(config)# no vlan group eng-dept vlan-list 30
Vlan 30 is successfully cleared from vlan group eng-dept.
Device(config)# show vlan group group-name eng-dept
```

This example shows how to clear all the VLAN groups:

```
Device(config)# no vlan group eng-dept vlan-list all
Device(config)# show vlan-group all
```

For more information about these commands, see the *Cisco IOS Security Command Reference*.

Configuring NAC Layer 2 802.1x Validation

You can configure NAC Layer 2 802.1x validation, which is also referred to as 802.1x authentication with a RADIUS server.

Beginning in privileged EXEC mode, follow these steps to configure NAC Layer 2 802.1x validation. The procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **switchport mode access**
4. **authentication event no-response action authorize vlan** *vlan-id*
5. **authentication periodic**
6. **authentication timer reauthenticate**
7. **end**
8. **show authentication sessions interface** *interface-id*
9. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode.
Step 2	interface <i>interface-id</i> Example: Device(config)# <code>interface gigabitethernet2/0/3</code>	Specifies the port to be configured, and enter interface configuration mode.
Step 3	switchport mode access Example: Device(config-if)# <code>switchport mode access</code>	Sets the port to access mode only if you configured the RADIUS server.
Step 4	authentication event no-response action authorize vlan <i>vlan-id</i> Example: Device(config-if)# <code>authentication event no-response action authorize vlan 8</code>	Specifies an active VLAN as an 802.1x guest VLAN. The range is 1 to 4094. You can configure any active VLAN except an internal VLAN (routed port), an RSPAN VLAN, or a voice VLAN as an 802.1x guest VLAN.
Step 5	authentication periodic Example: Device(config-if)# <code>authentication periodic</code>	Enables periodic re-authentication of the client, which is disabled by default.
Step 6	authentication timer reauthenticate Example: Device(config-if)# <code>authentication timer reauthenticate</code>	Sets re-authentication attempt for the client (set to one hour). This command affects the behavior of the switch only if periodic re-authentication is enabled.
Step 7	end Example: Device(config-if)# <code>end</code>	Returns to privileged EXEC mode.
Step 8	show authentication sessions interface <i>interface-id</i> Example:	Verifies your entries.

	Command or Action	Purpose
	Device# <code>show authentication sessions interface gigabitethernet2/0/3</code>	
Step 9	copy running-config startup-config Example: Device# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Configuring Limiting Login for Users

SUMMARY STEPS

1. `enable`
2. `configure terminal`
3. `aaa new-model`
4. `aaa authentication login default local`
5. `aaa authentication rejected n in m ban x`
6. `end`
7. `show aaa local user blocked`
8. `clear aaa local user blocked username username`

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> <code>enable</code>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode.
Step 3	aaa new-model Example: Device(config)# <code>aaa new-model</code>	Enables the authentication, authorization, and accounting (AAA) access control model.
Step 4	aaa authentication login default local Example: Device(config)# <code>aaa authentication login default local</code>	Sets the authentication, authorization, and accounting (AAA) authentication by using the default authentication methods.

	Command or Action	Purpose
Step 5	aaa authentication rejected <i>n</i> in <i>m</i> ban <i>x</i> Example: <pre>Device(config)# aaa authentication rejected 3 in 20 ban 300</pre>	Configures the time period for which an user is blocked, if the user fails to successfully login within the specified time and login attempts. <ul style="list-style-type: none"> • <i>n</i>—Specifies the number of times a user can try to login. • <i>m</i>—Specifies the number of seconds within which an user can try to login. • <i>x</i>—Specifies the time period an user is banned if the user fails to successfully login.
Step 6	end Example: <pre>Device(config)# end</pre>	Exits global configuration mode and returns to privileged EXEC mode.
Step 7	show aaa local user blocked Example: <pre>Device# show aaa local user blocked</pre>	Displays the list of local users who were blocked.
Step 8	clear aaa local user blocked username <i>username</i> Example: <pre>Device# clear aaa local user blocked username user1</pre>	Clears the information about the blocked local user.

Example

The following is sample output from the **show aaa local user blocked** command:

```
Device# show aaa local user blocked

Local-user          State
-----
user1               Watched (till 11:34:42 IST Feb 5 2015)
```

Configuring an Authenticator Switch with NEAT

Configuring this feature requires that one switch outside a wiring closet is configured as a supplicant and is connected to an authenticator switch.



Note

- The authenticator switch interface configuration must be restored to access mode by explicitly flapping it if a line card is removed and inserted in the chassis when CISP or NEAT session is active.
- The *cisco-av-pairs* must be configured as *device-traffic-class=switch* on the ISE, which sets the interface as a trunk after the supplicant is successfully authenticated.

Beginning in privileged EXEC mode, follow these steps to configure a switch as an authenticator:

SUMMARY STEPS

1. **configure terminal**
2. **cisp enable**
3. **interface *interface-id***
4. **switchport mode access**
5. **authentication port-control auto**
6. **dot1x pae authenticator**
7. **spanning-tree portfast**
8. **end**
9. **show running-config interface *interface-id***
10. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 2	cisp enable Example: Device(config)# cisp enable	Enables CISP.
Step 3	interface <i>interface-id</i> Example: Device(config)# interface gigabitethernet 2/0/1	Specifies the port to be configured, and enter interface configuration mode.
Step 4	switchport mode access Example: Device(config-if)# switchport mode access	Sets the port mode to access .
Step 5	authentication port-control auto Example: Device(config-if)# authentication port-control auto	Sets the port-authentication mode to auto .

	Command or Action	Purpose
Step 6	dot1x pae authenticator Example: <pre>Device(config-if)# dot1x pae authenticator</pre>	Configures the interface as a port access entity (PAE) authenticator.
Step 7	spanning-tree portfast Example: <pre>Device(config-if)# spanning-tree portfast trunk</pre>	Enables Port Fast on an access port connected to a single workstation or server..
Step 8	end Example: <pre>Device(config-if)# end</pre>	Returns to privileged EXEC mode.
Step 9	show running-config interface <i>interface-id</i> Example: <pre>Device# show running-config interface gigabitethernet 2/0/1</pre>	Verifies your configuration.
Step 10	copy running-config startup-config Example: <pre>Device# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file. Note Saving changes to the configuration file will mean that the authenticator interface will continue to be in trunk mode after reload. If you want the authenticator interface to remain as an access port, do not save your changes to the configuration file.

Configuring a Supplicant Switch with NEAT

Beginning in privileged EXEC mode, follow these steps to configure a switch as a supplicant:

SUMMARY STEPS

1. **configure terminal**
2. **cisp enable**
3. **dot1x credentials *profile***
4. **username *suppswitch***
5. **password *password***
6. **dot1x supplicant force-multicast**
7. **interface *interface-id***
8. **switchport trunk encapsulation dot1q**

9. **switchport mode trunk**
10. **dot1x pae supplicant**
11. **dot1x credentials** *profile-name*
12. **end**
13. **show running-config interface** *interface-id*
14. **copy running-config startup-config**
15. Configuring NEAT with Auto Smartports Macros

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 2	cisp enable Example: Device(config)# cisp enable	Enables CISP.
Step 3	dot1x credentials <i>profile</i> Example: Device(config)# dot1x credentials test	Creates 802.1x credentials profile. This must be attached to the port that is configured as supplicant.
Step 4	username <i>suppswitch</i> Example: Device(config)# username suppswitch	Creates a username.
Step 5	password <i>password</i> Example: Device(config)# password myswitch	Creates a password for the new username.
Step 6	dot1x supplicant force-multicast Example: Device(config)# dot1x supplicant force-multicast	Forces the switch to send only multicast EAPOL packets when it receives either unicast or multicast packets. This also allows NEAT to work on the supplicant switch in all host modes.
Step 7	interface <i>interface-id</i> Example:	Specifies the port to be configured, and enter interface configuration mode.

	Command or Action	Purpose
	Device(config)# <code>interface gigabitethernet1/0/1</code>	
Step 8	switchport trunk encapsulation dot1q Example: Device(config-if)# <code>switchport trunk encapsulation dot1q</code>	Sets the port to trunk mode.
Step 9	switchport mode trunk Example: Device(config-if)# <code>switchport mode trunk</code>	Configures the interface as a VLAN trunk port.
Step 10	dot1x pae supplicant Example: Device(config-if)# <code>dot1x pae supplicant</code>	Configures the interface as a port access entity (PAE) supplicant.
Step 11	dot1x credentials <i>profile-name</i> Example: Device(config-if)# <code>dot1x credentials test</code>	Attaches the 802.1x credentials profile to the interface.
Step 12	end Example: Device(config-if)# <code>end</code>	Returns to privileged EXEC mode.
Step 13	show running-config interface <i>interface-id</i> Example: Device# <code>show running-config interface gigabitethernet1/0/1</code>	Verifies your configuration.
Step 14	copy running-config startup-config Example: Device# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.
Step 15	Configuring NEAT with Auto Smartports Macros	You can also use an Auto Smartports user-defined macro instead of the switch VSA to configure the authenticator

	Command or Action	Purpose
		switch. For more information, see the <i>Auto Smartports Configuration Guide</i> for this release.

Configuring 802.1x Authentication with Downloadable ACLs and Redirect URLs



Note You must configure a downloadable ACL on the ACS before downloading it to the switch.

After authentication on the port, you can use the **show ip access-list** privileged EXEC command to display the downloaded ACLs on the port.

Configuring Downloadable ACLs

The policies take effect after client authentication and the client IP address addition to the IP device tracking table. The switch then applies the downloadable ACL to the port.

Beginning in privileged EXEC mode:

SUMMARY STEPS

1. **configure terminal**
2. **ip device tracking**
3. **aaa new-model**
4. **aaa authorization network default local group radius**
5. **radius-server vsa send authentication**
6. **interface *interface-id***
7. **ip access-group *acl-id* in**
8. **show running-config interface *interface-id***
9. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode.
Step 2	ip device tracking Example: Device(config)# <code>ip device tracking</code>	Sets the ip device tracking table.

	Command or Action	Purpose
Step 3	aaa new-model Example: <pre>Device(config)# aaa new-model</pre>	Enables AAA.
Step 4	aaa authorization network default local group radius Example: <pre>Device(config)# aaa authorization network default local group radius</pre>	Sets the authorization method to local. To remove the authorization method, use the no aaa authorization network default local group radius command.
Step 5	radius-server vsa send authentication Example: <pre>Device(config)# radius-server vsa send authentication</pre>	Configures the radius vsa send authentication.
Step 6	interface <i>interface-id</i> Example: <pre>Device(config)# interface gigabitethernet2/0/4</pre>	Specifies the port to be configured, and enter interface configuration mode.
Step 7	ip access-group <i>acl-id</i> in Example: <pre>Device(config-if)# ip access-group default_acl in</pre>	Configures the default ACL on the port in the input direction. Note The <i>acl-id</i> is an access list name or number.
Step 8	show running-config interface <i>interface-id</i> Example: <pre>Device(config-if)# show running-config interface gigabitethernet2/0/4</pre>	Verifies your configuration.
Step 9	copy running-config startup-config Example: <pre>Device# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Configuring a Downloadable Policy

Beginning in privileged EXEC mode:

SUMMARY STEPS

1. **configure terminal**
2. **access-list** *access-list-number* { **deny** | **permit** } { **hostname** | **any** | **host** } **log**
3. **interface** *interface-id*
4. **ip access-group** *acl-id* **in**
5. **exit**
6. **aaa new-model**
7. **aaa authorization network default group radius**
8. **ip device tracking**
9. **ip device tracking probe** [**count** | **interval** | **use-svi**]
10. **radius-server vsa send authentication**
11. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 2	access-list <i>access-list-number</i> { deny permit } { hostname any host } log Example: Device(config)# access-list 1 deny any log	Defines the default port ACL. The access-list-number is a decimal number from 1 to 99 or 1300 to 1999. Enter deny or permit to specify whether to deny or permit access if conditions are matched. The source is the source address of the network or host that sends a packet, such as this: <ul style="list-style-type: none"> • hostname: The 32-bit quantity in dotted-decimal format. • any: The keyword any as an abbreviation for source and source-wildcard value of 0.0.0.0 255.255.255.255. You do not need to enter a source-wildcard value. • host: The keyword host as an abbreviation for source and source-wildcard of source 0.0.0.0. (Optional) Applies the source-wildcard wildcard bits to the source. (Optional) Enters log to cause an informational logging message about the packet that matches the entry to be sent to the console.
Step 3	interface <i>interface-id</i> Example:	Enters interface configuration mode.

	Command or Action	Purpose
	Device(config)# <code>interface gigabitethernet2/0/2</code>	
Step 4	<p>ip access-group <i>acl-id</i> in</p> <p>Example:</p> <pre>Device(config-if)# ip access-group default_acl in</pre>	<p>Configures the default ACL on the port in the input direction.</p> <p>Note The <i>acl-id</i> is an access list name or number.</p>
Step 5	<p>exit</p> <p>Example:</p> <pre>Device(config-if)# exit</pre>	Returns to global configuration mode.
Step 6	<p>aaa new-model</p> <p>Example:</p> <pre>Device(config)# aaa new-model</pre>	Enables AAA.
Step 7	<p>aaa authorization network default group radius</p> <p>Example:</p> <pre>Device(config)# aaa authorization network default group radius</pre>	Sets the authorization method to local. To remove the authorization method, use the no aaa authorization network default group radius command.
Step 8	<p>ip device tracking</p> <p>Example:</p> <pre>Device(config)# ip device tracking</pre>	<p>Enables the IP device tracking table.</p> <p>To disable the IP device tracking table, use the no ip device tracking global configuration commands.</p>
Step 9	<p>ip device tracking probe [count interval use-svi]</p> <p>Example:</p> <pre>Device(config)# ip device tracking probe count</pre>	<p>(Optional) Configures the IP device tracking table:</p> <ul style="list-style-type: none"> • count <i>count</i>—Sets the number of times that the switch sends the ARP probe. The range is from 1 to 5. The default is 3. • interval <i>interval</i>—Sets the number of seconds that the switch waits for a response before resending the ARP probe. The range is from 30 to 300 seconds. The default is 30 seconds. • use-svi—Uses the switch virtual interface (SVI) IP address as source of ARP probes.
Step 10	<p>radius-server vsa send authentication</p> <p>Example:</p>	Configures the network access server to recognize and use vendor-specific attributes.

	Command or Action	Purpose
	Device (config) # radius-server vsa send authentication	Note The downloadable ACL must be operational.
Step 11	end Example: Device (config) # end	Returns to privileged EXEC mode.

Configuring VLAN ID-based MAC Authentication

Beginning in privileged EXEC mode, follow these steps:

SUMMARY STEPS

1. **configure terminal**
2. **mab request format attribute 32 vlan access-vlan**
3. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 2	mab request format attribute 32 vlan access-vlan Example: Device (config) # mab request format attribute 32 vlan access-vlan	Enables VLAN ID-based MAC authentication.
Step 3	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Configuring Flexible Authentication Ordering

The examples used in the instructions below changes the order of Flexible Authentication Ordering so that MAB is attempted before IEEE 802.1X authentication (dot1x). MAB is configured as the first authentication method, so MAB will have priority over all other authentication methods.

Beginning in privileged EXEC mode, follow these steps:

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **switchport mode access**
4. **authentication order** [dot1x | mab] | {webauth}
5. **authentication priority** [dot1x | mab] | {webauth}
6. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 2	interface <i>interface-id</i> Example: Device(config)# interface gigabitethernet 1/0/1	Specifies the port to be configured, and enter interface configuration mode.
Step 3	switchport mode access Example: Device(config-if)# switchport mode access	Sets the port to access mode only if you previously configured the RADIUS server.
Step 4	authentication order [dot1x mab] {webauth} Example: Device(config-if)# authentication order mab dot1x	(Optional) Sets the order of authentication methods used on a port.
Step 5	authentication priority [dot1x mab] {webauth} Example: Device(config-if)# authentication priority mab dot1x	(Optional) Adds an authentication method to the port-priority list.

	Command or Action	Purpose
Step 6	end Example: Device(config-if)# end	Returns to privileged EXEC mode.

Configuring Open1x

Beginning in privileged EXEC mode, follow these steps to enable manual control of the port authorization state:

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **switchport mode access**
4. **authentication control-direction** {both | in}
5. **authentication fallback** *name*
6. **authentication host-mode** [multi-auth | multi-domain | multi-host | single-host]
7. **authentication open**
8. **authentication order** [dot1x | mab] | {webauth}
9. **authentication periodic**
10. **authentication port-control** {auto | force-authorized | force-un authorized}
11. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 2	interface <i>interface-id</i> Example: Device(config)# interface gigabitethernet 1/0/1	Specifies the port to be configured, and enter interface configuration mode.
Step 3	switchport mode access Example: Device(config-if)# switchport mode access	Sets the port to access mode only if you configured the RADIUS server.

	Command or Action	Purpose
Step 4	authentication control-direction {both in} Example: <pre>Device(config-if) # authentication control-direction both</pre>	(Optional) Configures the port control as unidirectional or bidirectional.
Step 5	authentication fallback <i>name</i> Example: <pre>Device(config-if) # authentication fallback profile1</pre>	(Optional) Configures a port to use web authentication as a fallback method for clients that do not support 802.1x authentication.
Step 6	authentication host-mode [multi-auth multi-domain multi-host single-host] Example: <pre>Device(config-if) # authentication host-mode multi-auth</pre>	(Optional) Sets the authorization manager mode on a port.
Step 7	authentication open Example: <pre>Device(config-if) # authentication open</pre>	(Optional) Enables or disable open access on a port.
Step 8	authentication order [dot1x mab] {webauth} Example: <pre>Device(config-if) # authentication order dot1x webauth</pre>	(Optional) Sets the order of authentication methods used on a port.
Step 9	authentication periodic Example: <pre>Device(config-if) # authentication periodic</pre>	(Optional) Enables or disable reauthentication on a port.
Step 10	authentication port-control {auto force-authorized force-un authorized} Example: <pre>Device(config-if) # authentication port-control auto</pre>	(Optional) Enables manual control of the port authorization state.

	Command or Action	Purpose
Step 11	end Example: Device(config-if) # end	Returns to privileged EXEC mode.

Disabling 802.1x Authentication on the Port

You can disable 802.1x authentication on the port by using the **no dot1x pae** interface configuration command.

Beginning in privileged EXEC mode, follow these steps to disable 802.1x authentication on the port. This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **interface *interface-id***
3. **switchport mode access**
4. **no dot1x pae authenticator**
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 2	interface <i>interface-id</i> Example: Device(config) # interface gigabitethernet 2/0/1	Specifies the port to be configured, and enter interface configuration mode.
Step 3	switchport mode access Example: Device(config-if) # switchport mode access	(Optional) Sets the port to access mode only if you configured the RADIUS server.
Step 4	no dot1x pae authenticator Example: Device(config-if) # no dot1x pae authenticator	Disables 802.1x authentication on the port.

	Command or Action	Purpose
Step 5	end Example: Device(config-if) # end	Returns to privileged EXEC mode.

Resetting the 802.1x Authentication Configuration to the Default Values

Beginning in privileged EXEC mode, follow these steps to reset the 802.1x authentication configuration to the default values. This procedure is optional.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **dot1x default**
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 2	interface <i>interface-id</i> Example: Device(config)# interface gigabitethernet 1/0/2	Enters interface configuration mode, and specify the port to be configured.
Step 3	dot1x default Example: Device(config-if) # dot1x default	Resets the 802.1x parameters to the default values.
Step 4	end Example: Device(config-if) # end	Returns to privileged EXEC mode.

Monitoring 802.1x Statistics and Status

Table 147: Privileged EXEC show Commands

Command	Purpose
<code>show dot1x all statistics</code>	Displays 802.1x statistics for all ports
<code>show dot1x interface <i>interface-id</i> statistics</code>	Displays 802.1x statistics for a specific port
<code>show dot1x all [count details statistics summary]</code>	Displays the 802.1x administrative and operational status for a switch
<code>show dot1x interface <i>interface-id</i></code>	Displays the 802.1x administrative and operational status for a specific port

Table 148: Global Configuration Commands

Command	Purpose
<code>no dot1x logging verbose</code>	Filters verbose 802.1x authentication messages (beginning with Cisco IOS Release 12.2(55)SE)

For detailed information about the fields in these displays, see the command reference for this release.

Additional References for IEEE 802.1x Port-Based Authentication

Related Documents

Related Topic	Document Title
Configuring Identity Control policies and Identity Service templates for Session Aware networking.	Session Aware Networking Configuration Guide, Cisco IOS XE Release 3SE (Catalyst 3850 Switches) http://www.cisco.com/en/US/docs/ios-xml/ios/san/configuration/xe-3se/3850/san-xe-3se-3850-book.html

Related Topic	Document Title
Configuring RADIUS, TACACS+, Secure Shell, 802.1X and AAA.	Securing User Services Configuration Guide Library, Cisco IOS XE Release 3SE (Catalyst 3850 Switches) http://www.cisco.com/en/US/docs/ios-xml/ios/security/config_library/xe-3se/3850/secuser-xe-3se-3850-1

Error Message Decoder

Description	Link
To help you research and resolve system error messages in this release, use the Error Message Decoder tool.	https://www.cisco.com/cgi-bin/Support/Errordecoder/index.cgi

MIBs

MIB	MIBs Link
All the supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
<p>The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies.</p> <p>To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds.</p> <p>Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.</p>	http://www.cisco.com/support

Feature Information for 802.1x Port-Based Authentication

Release	Feature Information
Cisco IOS Release 15.0(2)EX1	This feature was introduced.

Release	Feature Information
	Supports the use of same authorization methods on all the Catalyst switches in a network.
	Supports filtering verbose system messages from the authentication manager.



CHAPTER 75

Configuring Web-Based Authentication

The Web-Based Authentication feature, also known as web authentication proxy, authenticates end users on host systems that do not run the IEEE 802.1x supplicant.

- [Information About Web-Based Authentication, on page 1539](#)
- [How to Configure Web-Based Authentication, on page 1554](#)
- [Configuration Examples for Web-Based Authentication, on page 1570](#)
- [Additional References for Web-Based Authentication, on page 1572](#)
- [Feature Information for Web-Based Authentication, on page 1572](#)

Information About Web-Based Authentication

Web-Based Authentication Overview

Use the web-based authentication feature, known as web authentication proxy, to authenticate end users on host systems that do not run the IEEE 802.1x supplicant.



Note You can configure web-based authentication on Layer 2 and Layer 3 interfaces.

When you initiate an HTTP session, web-based authentication intercepts ingress HTTP packets from the host and sends an HTML login page to the users. The users enter their credentials, which the web-based authentication feature sends to the authentication, authorization, and accounting (AAA) server for authentication.

If authentication succeeds, web-based authentication sends a Login-Successful HTML page to the host and applies the access policies returned by the AAA server.

If authentication fails, web-based authentication forwards a Login-Fail HTML page to the user, prompting the user to retry the login. If the user exceeds the maximum number of attempts, web-based authentication forwards a Login-Expired HTML page to the host, and the user is placed on a watch list for a waiting period.



Note HTTPS traffic interception for central web authentication redirect is not supported.



Note You should use global parameter-map (for method-type, custom, and redirect) only for using the same web authentication methods like consent, web consent, and webauth, for all the clients and SSIDs. This ensures that all the clients have the same web-authentication method.

If the requirement is to use Consent for one SSID and Web-authentication for another SSID, then you should use two named parameter-maps. You should configure Consent in first parameter-map and configure webauth in second parameter-map.



Note The traceback that you receive when webauth client tries to do authentication does not have any performance or behavioral impact. It happens rarely when the context for which FFM replied back to EPM for ACL application is already dequeued (possibly due to timer expiry) and the session becomes ‘unauthorized’.

Based on where the web pages are hosted, the local web authentication can be categorized as follows:

- *Internal*—The internal default HTML pages (Login, Success, Fail, and Expire) in the controller are used during the local web authentication.
- *Customized*—The customized web pages (Login, Success, Fail, and Expire) are downloaded onto the controller and used during the local web authentication.
- *External*—The customized web pages are hosted on the external web server instead of using the in-built or custom web pages.

Based on the various web authentication pages, the types of web authentication are as follows:

- *Webauth*—This is a basic web authentication. Herein, the controller presents a policy page with the user name and password. You need to enter the correct credentials to access the network.
- *Consent or web-passthrough*—Herein, the controller presents a policy page with the Accept or Deny buttons. You need to click the Accept button to access the network.
- *Webconsent*—This is a combination of webauth and consent web authentication types. Herein, the controller presents a policy page with Accept or Deny buttons along with user name or password. You need to enter the correct credentials and click the Accept button to access the network.

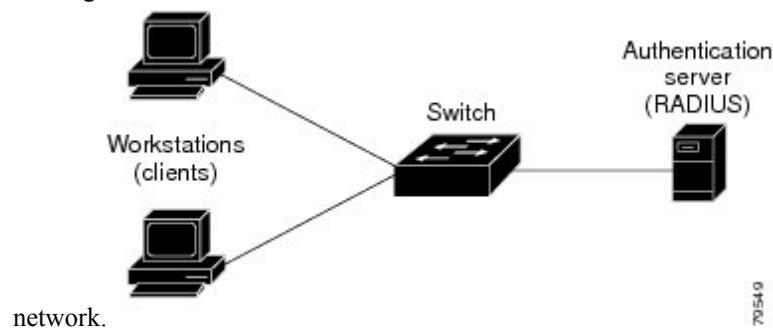
Device Roles

With web-based authentication, the devices in the network have these specific roles:

- *Client*—The device (workstation) that requests access to the LAN and the services and responds to requests from the switch. The workstation must be running an HTML browser with Java Script enabled.
- *Authentication server*—Authenticates the client. The authentication server validates the identity of the client and notifies the switch that the client is authorized to access the LAN and the switch services or that the client is denied.
- *Switch*—Controls the physical access to the network based on the authentication status of the client. The switch acts as an intermediary (proxy) between the client and the authentication server, requesting identity information from the client, verifying that information with the authentication server, and relaying a response to the client.

Figure 114: Web-Based Authentication Device Roles

This figure shows the roles of these devices in a



Host Detection

The switch maintains an IP device tracking table to store information about detected hosts.



Note By default, the IP device tracking feature is disabled on a switch. You must enable the IP device tracking feature to use web-based authentication.

For Layer 2 interfaces, web-based authentication detects IP hosts by using these mechanisms:

- ARP based trigger—ARP redirect ACL allows web-based authentication to detect hosts with a static IP address or a dynamic IP address.
- Dynamic ARP inspection
- DHCP snooping—Web-based authentication is notified when the switch creates a DHCP-binding entry for the host.

Session Creation

When web-based authentication detects a new host, it creates a session as follows:

- Reviews the exception list.
If the host IP is included in the exception list, the policy from the exception list entry is applied, and the session is established.
- Reviews for authorization bypass
If the host IP is not on the exception list, web-based authentication sends a nonresponsive-host (NRH) request to the server.
If the server response is access accepted, authorization is bypassed for this host. The session is established.
- Sets up the HTTP intercept ACL
If the server response to the NRH request is access rejected, the HTTP intercept ACL is activated, and the session waits for HTTP traffic from the host.

Authentication Process

When you enable web-based authentication, these events occur:

- The user initiates an HTTP session.
- The HTTP traffic is intercepted, and authorization is initiated. The switch sends the login page to the user. The user enters a username and password, and the switch sends the entries to the authentication server.
- If the authentication succeeds, the switch downloads and activates the user's access policy from the authentication server. The login success page is sent to the user.
- If the authentication fails, the switch sends the login fail page. The user retries the login. If the maximum number of attempts fails, the switch sends the login expired page, and the host is placed in a watch list. After the watch list times out, the user can retry the authentication process.
- If the authentication server does not respond to the switch, and if an AAA fail policy is configured, the switch applies the failure access policy to the host. The login success page is sent to the user.
- The switch reauthenticates a client when the host does not respond to an ARP probe on a Layer 2 interface, or when the host does not send any traffic within the idle timeout on a Layer 3 interface.
- The feature applies the downloaded timeout or the locally configured session timeout.



Note Beginning with Cisco IOS XE Denali 16.1.1 and later, the default session timeout value for web-based authentication on WLC is 1800 seconds. The default session timeout value was infinite seconds, prior to Cisco IOS XE Denali 16.1.1.

- If the terminate action is RADIUS, the feature sends a nonresponsive host (NRH) request to the server. The terminate action is included in the response from the server.
- If the terminate action is default, the session is dismantled, and the applied policy is removed.

Using Authentication Proxy

The authentication proxy feature requires some user interaction on the client host. The table below describes the interaction of the authentication proxy with the client host.

Table 149: Authentication Proxy Interaction with the Client Host

Authentication Proxy Action with Client	Description
Triggering on HTTP connections	If a user is not currently authenticated at the firewall router, any HTTP connection initiated by the user triggers the authentication proxy. If the user is already authenticated, the authentication proxy is transparent to the user.
Logging in using the login page	Triggering the authentication proxy generates an HTML-based login page. The user must enter a username and password to be authenticated with the AAA server. The Authentication Proxy Login Page figure, in the How the Authentication Proxy Works module, illustrates the authentication proxy login page.

Authentication Proxy Action with Client	Description
Authenticating the user at the client	<p>Following the login attempt, the authentication proxy action can vary depending on whether JavaScript is enabled in the browser. If JavaScript is enabled, and authentication is successful, the authentication proxy displays a message indicating the status of the authentication as shown in the Authentication Proxy Login Status Message figure, in the How the Authentication Proxy Works module. After the authentication status is displayed, the proxy automatically completes the HTTP connection.</p> <p>If JavaScript is disabled, and authentication is successful, the authentication proxy generates a popup window with additional instructions for completing the connection. See the Authentication Proxy Login Status Message with JavaScript Disabled figure, in the Secure Authentication module.</p> <p>If authentication is unsuccessful in any case, the user must log in again from the login page.</p>

When to Use the Authentication Proxy

The following are some situations in which you can use the authentication proxy:

- You want to manage access privileges on an individual (per-user) basis using the services provided by the authentication servers instead of configuring access control based on host IP address or global access policies. Authenticating and authorizing users from any host IP address also allows network administrators to configure host IP addresses using DHCP.
- You want to authenticate and authorize local users before permitting access to intranet or Internet services.
- You want to authenticate and authorize remote users before permitting access to local services.
- You want to control access for specific extranet users. For example, you might want to authenticate and authorize the financial officer of a corporate partner with one set of access privileges while authorizing the technology officer for that same partner to use another set of access privileges.
- You want to use the authentication proxy in conjunction with VPN client software to validate users and to assign specific access privileges.
- You want to use the authentication proxy in conjunction with AAA accounting to generate “start” and “stop” accounting records that can be used for billing, security, or resource allocation purposes, thereby allowing users to track traffic from the authenticated hosts.

Applying Authentication Proxy

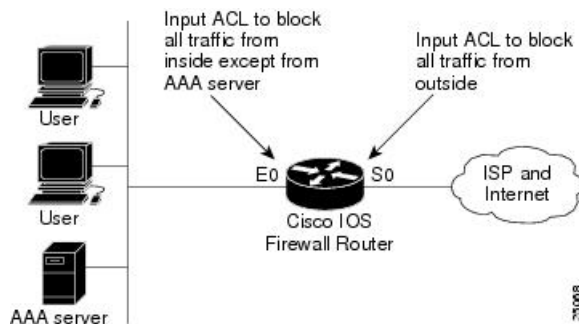
Apply the authentication proxy in the inbound direction at any interface on the router where you want per-user authentication and authorization. Applying the authentication proxy inbound at an interface causes it to intercept the initial connection request from an user, before that request is subjected to any other processing. If the user fails to gain authentication with the AAA server, the connection request is dropped.

How you apply the authentication proxy depends on your security policy. For example, you can block all traffic through an interface and enable the authentication proxy feature to require authentication and authorization for all user-initiated HTTP connections. Users are authorized for services only after successful authentication with the AAA server.

The authentication proxy feature also allows you to use standard access lists to specify a host or group of hosts whose initial HTTP traffic triggers the proxy.

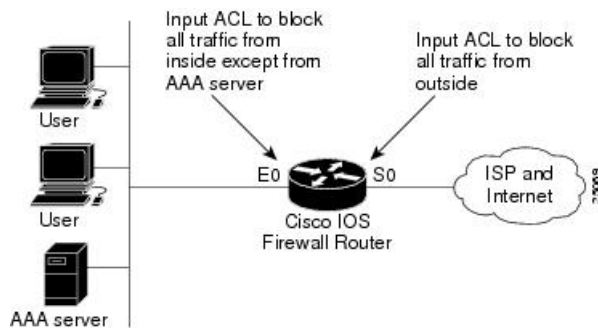
The figure below shows the authentication proxy applied at the LAN interface with all network users required to be authenticated upon the initial connection (all traffic is blocked at each interface).

Figure 115: Applying the Authentication Proxy at the Local Interface



The figure below shows the authentication proxy applied at the dial-in interface with all network traffic blocked at each interface.

Figure 116: Applying the Authentication Proxy at an Outside Interface



Local Web Authentication Banner

With Web Authentication, you can create a default and customized web-browser banners that appears when you log in to a switch.

The banner appears on both the login page and the authentication-result pop-up pages. The default banner messages are as follows:

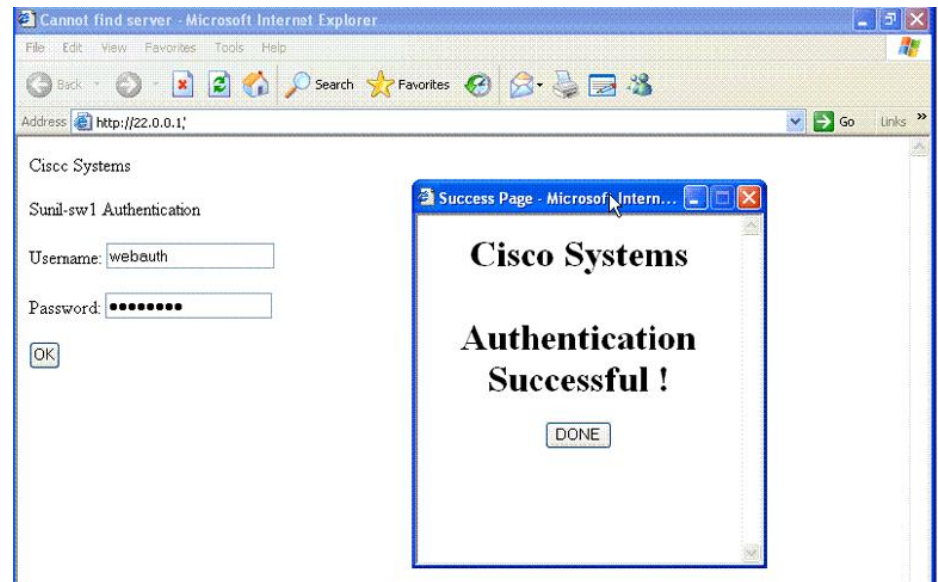
- *Authentication Successful*
- *Authentication Failed*
- *Authentication Expired*

The Local Web Authentication Banner can be configured in legacy and new-style (Session-aware) CLIs as follows:

- Legacy mode—Use the **ip admission auth-proxy-banner http** global configuration command.
- New-style mode—Use the **parameter-map type webauth global banner** global configuration command.

The default banner *Cisco Systems* and *Switch host-name Authentication* appear on the Login Page. *Cisco Systems* appears on the authentication result pop-up page.

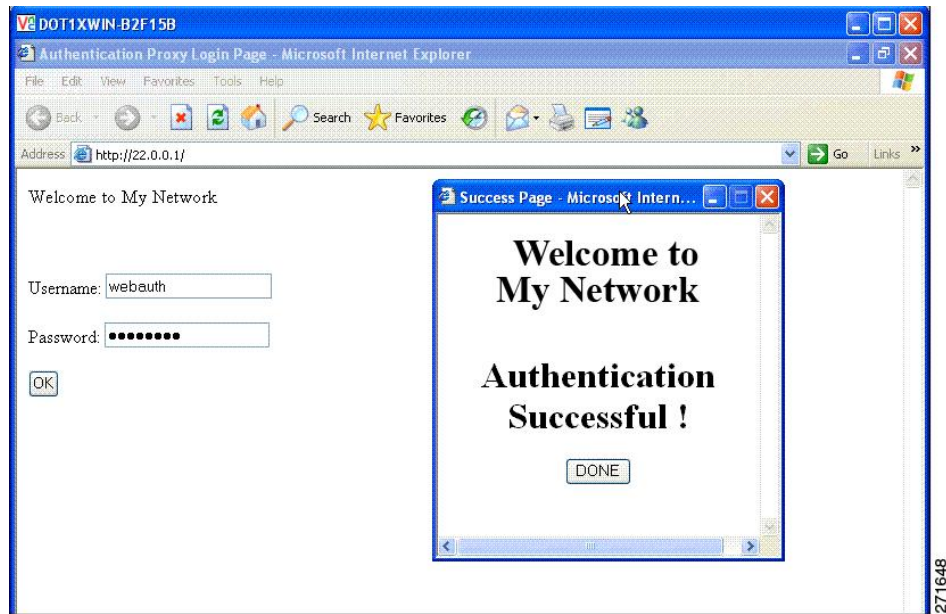
Figure 117: Authentication Successful Banner



The banner can be customized as follows:

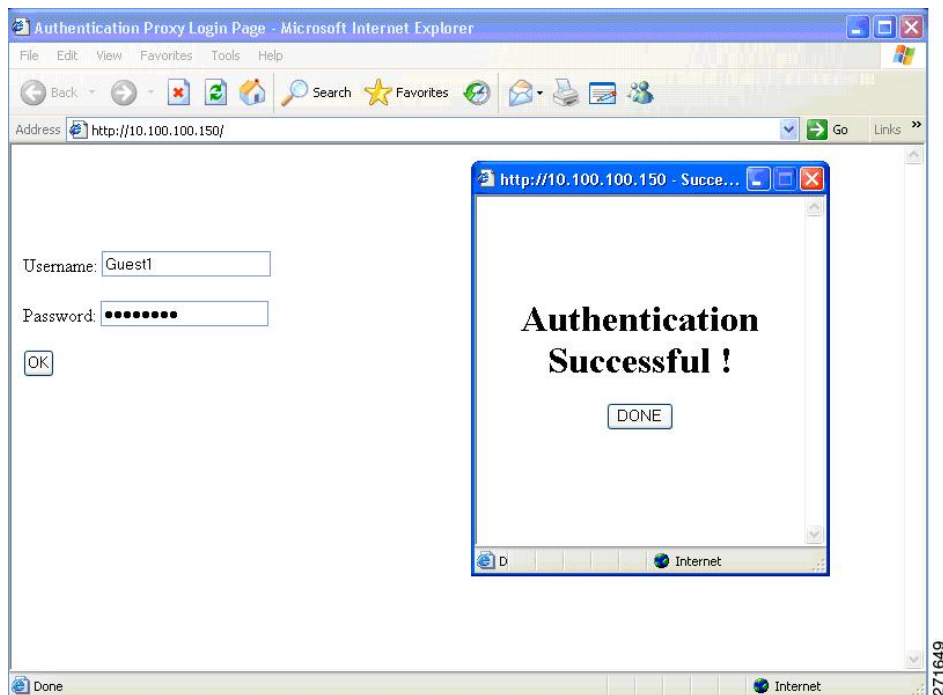
- Add a message, such as switch, router, or company name to the banner:
 - Legacy mode—Use the **ip admission auth-proxy-banner http banner-text** global configuration command.
 - New-style mode—Use the **parameter-map type webauth global banner** global configuration command.
- Add a logo or text file to the banner:
 - Legacy mode—Use the **ip admission auth-proxy-banner http file-path** global configuration command.
 - New-style mode—Use the **parameter-map type webauth global banner** global configuration command.

Figure 118: Customized Web Banner



If you do not enable a banner, only the username and password dialog boxes appear in the web authentication login screen, and no banner appears when you log into the switch.

Figure 119: Login Screen With No Banner



Web Authentication Customizable Web Pages

During the web-based authentication process, the switch internal HTTP server hosts four HTML pages to deliver to an authenticating client. The server uses these pages to notify you of these four-authentication process states:

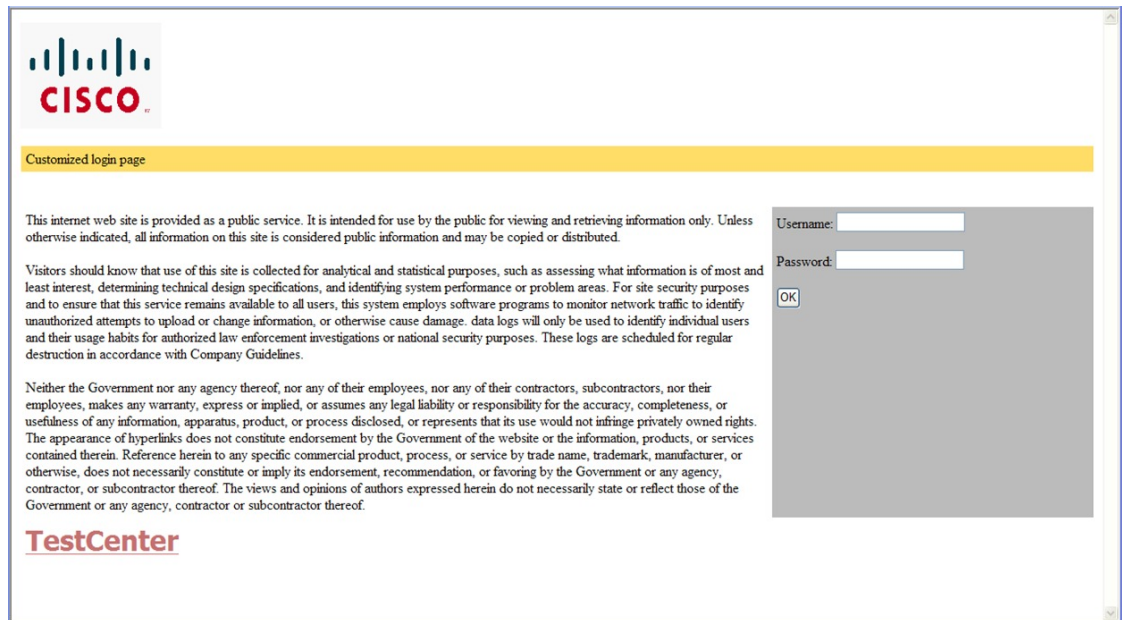
- Login—Your credentials are requested.
- Success—The login was successful.
- Fail—The login failed.
- Expire—The login session has expired because of excessive login failures.

Guidelines

- You can substitute your own HTML pages for the default internal HTML pages.
- You can use a logo or specify text in the *login*, *success*, *failure*, and *expire* web pages.
- On the banner page, you can specify text in the login page.
- The pages are in HTML.
- You must include an HTML redirect command in the success page to access a specific URL.
- The URL string must be a valid URL (for example, <http://www.cisco.com>). An incomplete URL might cause *page not found* or similar errors on a web browser.
- If you configure web pages for HTTP authentication, they must include the appropriate HTML commands (for example, to set the page time out, to set a hidden password, or to confirm that the same page is not submitted twice).
- The CLI command to redirect users to a specific URL is not available when the configured login form is enabled. The administrator should ensure that the redirection is configured in the web page.
- If the CLI command redirecting users to specific URL after authentication occurs is entered and then the command configuring web pages is entered, the CLI command redirecting users to a specific URL does not take effect.
- Configured web pages can be copied to the switch boot flash or flash.
- On stackable switches, configured pages can be accessed from the flash on the stack master or members.
- The login page can be on one flash, and the success and failure pages can be another flash (for example, the flash on the stack master or a member).
- You must configure all four pages.
- The banner page has no effect if it is configured with the web page.
- All of the logo files (image, flash, audio, video, and so on) that are stored in the system directory (for example, flash, disk0, or disk) and that must be displayed on the login page must use *web_auth_<filename>* as the file name.
- The configured authentication proxy feature supports both HTTP and SSL.

You can substitute your HTML pages for the default internal HTML pages. You can also specify a URL to which users are redirected after authentication occurs, which replaces the internal Success page.

Figure 120: Customizable Authentication Page



Authentication Proxy Web Page Guidelines

When configuring customized authentication proxy web pages, follow these guidelines:

- To enable the custom web pages feature, specify all four custom HTML files. If you specify fewer than four files, the internal default HTML pages are used.
- The four custom HTML files must be present on the flash memory of the switch. The maximum size of each HTML file is 8 KB.
- Any images on the custom pages must be on an accessible HTTP server. Configure an intercept ACL within the admission rule.
- Any external link from a custom page requires configuration of an intercept ACL within the admission rule.
- To access a valid DNS server, any name resolution required for external links or images requires configuration of an intercept ACL within the admission rule.
- If the custom web pages feature is enabled, a configured auth-proxy-banner is not used.
- If the custom web pages feature is enabled, the redirection URL for successful login feature is not available.
- To remove the specification of a custom file, use the **no** form of the command.

Because the custom login page is a public web form, consider these guidelines for the page:

- The login form must accept user entries for the username and password and must show them as **uname** and **pwd**.
- The custom login page should follow best practices for a web form, such as page timeout, hidden password, and prevention of redundant submissions.

Redirection URL for Successful Login Guidelines

When configuring a redirection URL for successful login, consider these guidelines:

- If the custom authentication proxy web pages feature is enabled, the redirection URL feature is disabled and is not available in the CLI. You can perform redirection in the custom-login success page.
- If the redirection URL feature is enabled, a configured auth-proxy-banner is not used
- To remove the specification of a redirection URL, use the **no** form of the command.
- If the redirection URL is required after the web-based authentication client is successfully authenticated, then the URL string must start with a valid URL (for example, http://) followed by the URL information. If only the URL is given without http://, then the redirection URL on successful authentication might cause page not found or similar errors on a web browser.

Web Authentication Redirection to Original URL Overview

The Web Authentication Redirection to Original URL feature enables networks to redirect guest users to the URL that they had originally requested. This feature is enabled by default and requires no configuration.

Guest networks are network connections provided by an enterprise to allow their guests to gain access to the Internet and to their own enterprise networks without compromising the security of the host enterprise. Guest users of an enterprise network can connect to the guest access network through either a wired Ethernet connection or a wireless connection.

Guest access uses a captive portal to gather all web requests made by guests and redirect these requests to one of the guest on-boarding web pages. When guests successfully complete the guest workflow, they are redirected to the page that they had originally requested.

The originally requested URL is passed as metadata along with the Cisco Identity Services Engine (ISE) guest access redirect URL. The Cisco ISE is a security policy management and control platform. It automates and simplifies access control and security compliance for wired, wireless, and VPN connectivity. The requested URL is added at the end of the Cisco ISE guest URL so that the device can send the redirect URL to the guest client. The Cisco ISE parses the URL and redirects the guest to the original URL after completing the on-boarding.

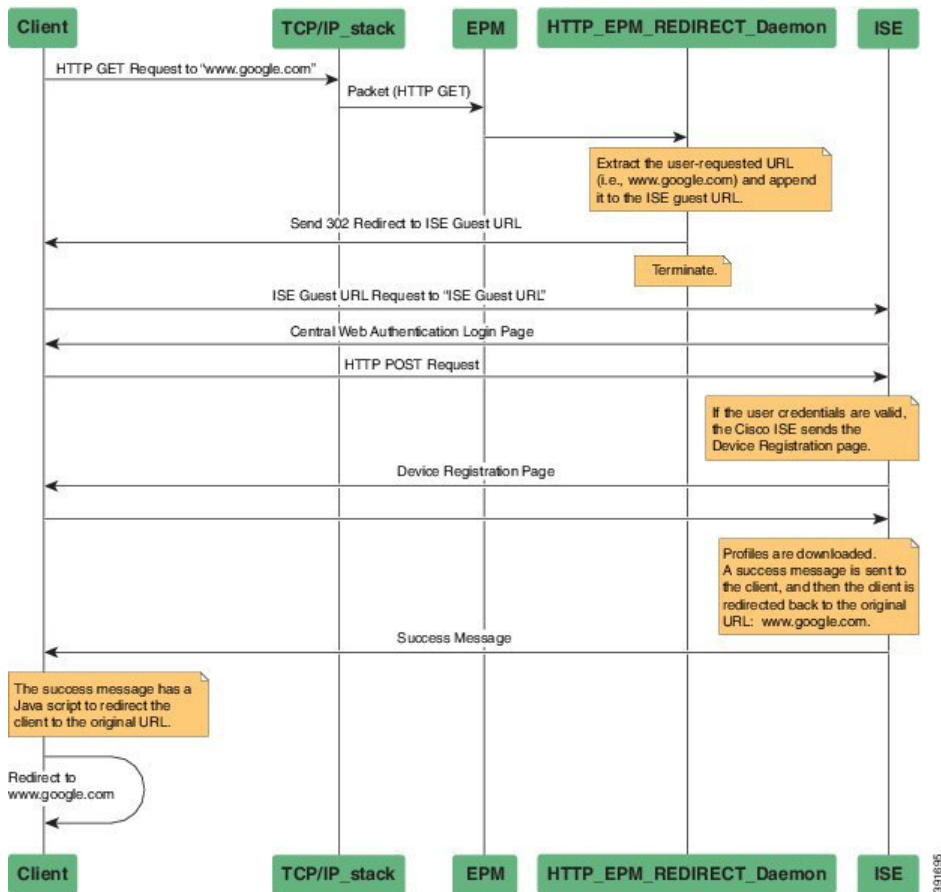
The following is an example of a redirect URL along with the original requested URL:

```
https://10.64.67.92:8443/guestportal/gateway?sessionId=0920269E0000000B0002426B&action=cwa&redirect_url=http://www.cisco.com/
```

In this example, the URL, `https://10.64.67.92:8443/guestportal/gateway?sessionId=0920269E0000000B0002426B&action=cwa` is the URL for the guest portal, “&” tells the browser that what follows is a list of name value pairs, and `redirect_url=http://www.cisco.com` identifies the URL that the user originally requested and to which the user is redirected after completing the guest workflow.

This illustration displays the packet flow that redirects a user to the originally requested URL:

Figure 121: Original URL Redirection Packet Flow



1. A user accesses a network for the first time and sends an HTTP request to access www.google.com. When the user first accesses the network, a MAC authentication bypass (MAB) is triggered and the MAC address is sent to the Cisco ISE.
2. The Cisco ISE returns a RADIUS access-accept message (even if the MAC address is not received) along with the redirect access control list (ACL), the ACL-WEBAUTH-REDIRECT message, and the guest web portal URL to the device.

The RADIUS message instructs the device to open a port that is restricted based on the configured port and the redirect ACLs, for regular network traffic.

3. When the user launches a web browser, the device intercepts the HTTP traffic and redirects the browser to the Cisco ISE central web authentication (CWA) guest web portal URL; the user-requested URL is extracted and appended to the Cisco ISE guest URL.
4. When the user is authenticated, the Cisco ISE sends the Device Registration page to the user. The user enters the required information, and the page is returned to the Cisco ISE. The Cisco ISE downloads user profiles and redirects the user to the originally requested URL: www.google.com.

Web-based Authentication Interactions with Other Features

802.1x Authentication

These are the 802.1x authentication configuration guidelines:

- When 802.1x authentication is enabled, ports are authenticated before any other Layer 2 or Layer 3 features are enabled.
- If the VLAN to which an 802.1x-enabled port is assigned changes, this change is transparent and does not affect the switch. For example, this change occurs if a port is assigned to a RADIUS server-assigned VLAN and is then assigned to a different VLAN after re-authentication.

If the VLAN to which an 802.1x port is assigned to shut down, disabled, or removed, the port becomes unauthorized. For example, the port is unauthorized after the access VLAN to which a port is assigned shuts down or is removed.

- The 802.1x protocol is supported on Layer 2 static-access ports, voice VLAN ports, and Layer 3 routed ports, but it is not supported on these port types:
 - Dynamic ports—A port in dynamic mode can negotiate with its neighbor to become a trunk port. If you try to enable 802.1x authentication on a dynamic port, an error message appears, and 802.1x authentication is not enabled. If you try to change the mode of an 802.1x-enabled port to dynamic, an error message appears, and the port mode is not changed.
 - EtherChannel port—Do not configure a port that is an active or a not-yet-active member of an EtherChannel as an 802.1x port. If you try to enable 802.1x authentication on an EtherChannel port, an error message appears, and 802.1x authentication is not enabled.
 - Switched Port Analyzer (SPAN) and Remote SPAN (RSPAN) destination ports—You can enable 802.1x authentication on a port that is a SPAN or RSPAN destination port. However, 802.1x authentication is disabled until the port is removed as a SPAN or RSPAN destination port. You can enable 802.1x authentication on a SPAN or RSPAN source port.
- Before globally enabling 802.1x authentication on a switch by entering the **dot1x system-auth-control** global configuration command, remove the EtherChannel configuration from the interfaces on which 802.1x authentication and EtherChannel are configured.
- Cisco IOS Release 12.2(55)SE and later supports filtering of system messages related to 802.1x authentication.

AAA Accounting with Authentication Proxy

Using the authentication proxy, you can generate “start” and “stop” accounting records with enough information to be used for billing and security auditing purposes. Thus, you can monitor the actions of authenticated hosts that use the authentication proxy service.

When an authentication proxy cache and associated dynamic access control lists (ACLs) are created, the authentication proxy will start to track the traffic from the authenticated host. Accounting saves data about this event in a data structure stored with the data of other users. If the accounting start option is enabled, you can generate an accounting record (a “start” record) at this time. Subsequent traffic from the authenticated host will be recorded when the dynamic ACL created by the authentication proxy receives the packets.

When an authentication proxy cache expires and is deleted, additional data, such as elapsed time, is added to the accounting information and a “stop” record is sent to the server. At this point, the information is deleted from the data structure.

The accounting records for the authentication proxy user session are related to the cache and the dynamic ACL usage.

ACLs

If you configure a VLAN ACL or a Cisco IOS ACL on an interface, the ACL is applied to the host traffic only after the web-based authentication host policy is applied.

For Layer 2 web-based authentication, it is more secure, though not required, to configure a port ACL (PACL) as the default access policy for ingress traffic from hosts connected to the port. After authentication, the web-based authentication host policy overrides the PACL. The Policy ACL is applied to the session even if there is no ACL configured on the port.

You cannot configure a MAC ACL and web-based authentication on the same interface.

You cannot configure web-based authentication on a port whose access VLAN is configured for VACL capture.

Context-Based Access Control

Web-based authentication cannot be configured on a Layer 2 port if context-based access control (CBAC) is configured on the Layer 3 VLAN interface of the port VLAN.

EtherChannel

You can configure web-based authentication on a Layer 2 EtherChannel interface. The web-based authentication configuration applies to all member channels.

Gateway IP

You cannot configure Gateway IP (GWIP) on a Layer 3 VLAN interface if web-based authentication is configured on any of the switch ports in the VLAN.

You can configure web-based authentication on the same Layer 3 interface as Gateway IP. The host policies for both features are applied in software. The GWIP policy overrides the web-based authentication host policy.

LAN Port IP

You can configure LAN port IP (LPIP) and Layer 2 web-based authentication on the same port. The host is authenticated by using web-based authentication first, followed by LPIP posture validation. The LPIP host policy overrides the web-based authentication host policy.

If the web-based authentication idle timer expires, the NAC policy is removed. The host is authenticated, and posture is validated again.

Port Security

You can configure web-based authentication and port security on the same port. Web-based authentication authenticates the port, and port security manages network access for all MAC addresses, including that of the client. You can then limit the number or group of clients that can access the network through the port.

Default Web-Based Authentication Configuration

The following table shows the default web-based authentication configuration.

Table 150: Default Web-based Authentication Configuration

Feature	Default Setting
AAA	Disabled
RADIUS server <ul style="list-style-type: none"> • IP address • UDP authentication port • Key 	<ul style="list-style-type: none"> • None specified • 1645 • None specified
Default value of inactivity timeout	3600 seconds
Inactivity timeout	Enabled

Web-Based Authentication Configuration Guidelines and Restrictions

- Web-based authentication is an ingress-only feature.
- You can configure web-based authentication only on access ports. Web-based authentication is not supported on trunk ports, EtherChannel member ports, or dynamic trunk ports.
- External web authentication, where the switch redirects a client to a particular host or web server for displaying login message, is not supported.
- You must configure the default ACL on the interface before configuring web-based authentication. Configure a port ACL for a Layer 2 interface or a Cisco IOS ACL for a Layer 3 interface.
- You cannot authenticate hosts on Layer 2 interfaces with static ARP cache assignment. These hosts are not detected by the web-based authentication feature because they do not send ARP messages.
- By default, the IP device tracking feature is disabled on a switch. You must enable the IP device tracking feature to use web-based authentication.
- You must configure at least one IP address to run the switch HTTP server. You must also configure routes to reach each host IP address. The HTTP server sends the HTTP login page to the host.
- Hosts that are more than one hop away might experience traffic disruption if an STP topology change results in the host traffic arriving on a different port. This occurs because the ARP and DHCP updates might not be sent after a Layer 2 (STP) topology change.
- Web-based authentication does not support VLAN assignment as a downloadable-host policy.
- Web-based authentication supports IPv6 in Session-aware policy mode. IPv6 Web-authentication requires at least one IPv6 address configured on the switch and IPv6 Snooping configured on the switchport.
- Web-based authentication and Network Edge Access Topology (NEAT) are mutually exclusive. You cannot use web-based authentication when NEAT is enabled on an interface, and you cannot use NEAT when web-based authentication is running on an interface.

- Web-based authentication NRH (Non-Responsive Host) is not supported for voice devices.
- Only the Password Authentication Protocol (PAP) is supported for web-based RADIUS authentication on controllers. The Challenge Handshake Authentication Protocol (CHAP) is not supported for web-based RADIUS authentication on controllers.
- Identify the following RADIUS security server settings that will be used while configuring switch-to-RADIUS-server communication:
 - Host name
 - Host IP address
 - Host name and specific UDP port numbers
 - IP address and specific UDP port numbers

The combination of the IP address and UDP port number creates a unique identifier, that enables RADIUS requests to be sent to multiple UDP ports on a server at the same IP address. If two different host entries on the same RADIUS server are configured for the same service (for example, authentication) the second host entry that is configured functions as the failover backup to the first one. The RADIUS host entries are chosen in the order that they were configured.

- When you configure the RADIUS server parameters:
 - Specify the **key string** on a separate command line.
 - For **key string**, specify the authentication and encryption key used between the switch and the RADIUS daemon running on the RADIUS server. The key is a text string that must match the encryption key used on the RADIUS server.
 - When you specify the **key string**, use spaces within and at the end of the key. If you use spaces in the key, do not enclose the key in quotation marks unless the quotation marks are part of the key. This key must match the encryption used on the RADIUS daemon.
 - You can globally configure the timeout, retransmission, and encryption key values for all RADIUS servers by using with the **radius-server host** global configuration command. If you want to configure these options on a per-server basis, use the **radius-server timeout**, **radius-server transmit**, and the **radius-server key** global configuration commands.



Note You need to configure some settings on the RADIUS server, including: the switch IP address, the key string to be shared by both the server and the switch, and the downloadable ACL (DACL). For more information, see the RADIUS server documentation.

How to Configure Web-Based Authentication

Configuring the Authentication Rule and Interfaces

Follow these steps to configure the authentication rule and interfaces:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip admission name *name* proxy http**
4. **interface *type slot/port***
5. **ip access-group *name***
6. **ip admission name**
7. **exit**
8. **ip device tracking**
9. **end**
10. **show ip admission status**
11. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	ip admission name <i>name</i> proxy http Example: Device(config)# ip admission name webauth1 proxy http	Configures an authentication rule for web-based authorization.
Step 4	interface <i>type slot/port</i> Example: Device(config)# interface gigabitethernet 1/0/1	Enters interface configuration mode and specifies the ingress Layer 2 or Layer 3 interface to be enabled for web-based authentication. <i>type</i> can be fastethernet, gigabit ethernet, or tengigabitethernet.
Step 5	ip access-group <i>name</i> Example: Device(config-if)# ip access-group webauthag	Applies the default ACL.

	Command or Action	Purpose
Step 6	ip admission name Example: Device(config)# ip admission name	Configures an authentication rule for web-based authorization for the interface.
Step 7	exit Example: Device(config-if)# exit	Returns to configuration mode.
Step 8	ip device tracking Example: Device(config)# ip device tracking	Enables the IP device tracking table.
Step 9	end Example: Device(config)# end	Returns to privileged EXEC mode.
Step 10	show ip admission status Example: Device# show ip admission status	Displays the configuration.
Step 11	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Configuring AAA Authentication

SUMMARY STEPS

1. **aaa new-model**
2. **aaa authentication login default group {tacacs+ | radius}**
3. **aaa authorization auth-proxy default group {tacacs+ | radius}**
4. **tacacs-server host {hostname | ip_address}**
5. **tacacs-server key {key-data}**

DETAILED STEPS

	Command or Action	Purpose
Step 1	aaa new-model Example: <pre>Device(config)# aaa new-model</pre>	Enables AAA functionality.
Step 2	aaa authentication login default group {tacacs+ radius} Example: <pre>Device(config)# aaa authentication login default group tacacs+</pre>	Defines the list of authentication methods at login. named_authentication_list refers to any name that is not greater than 31 characters. AAA_group_name refers to the server group name. You need to define the server-group server_name at the beginning itself.
Step 3	aaa authorization auth-proxy default group {tacacs+ radius} Example: <pre>Device(config)# aaa authorization auth-proxy default group tacacs+</pre>	Creates an authorization method list for web-based authorization.
Step 4	tacacs-server host {hostname ip_address} Example: <pre>Device(config)# tacacs-server host 10.1.1.1</pre>	Specifies an AAA server.
Step 5	tacacs-server key {key-data} Example: <pre>Device(config)# tacacs-server key</pre>	Configures the authorization and encryption key used between the switch and the TACACS server.

Configuring Switch-to-RADIUS-Server Communication

Follow these steps to configure the RADIUS server parameters:

SUMMARY STEPS

1. enable
2. configure terminal
3. ip radius source-interface vlan *vlan interface number*
4. radius-server host {hostname | ip-address} test username *username*
5. radius-server key *string*
6. radius-server dead-criteria tries *num-tries*

7. end

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	ip radius source-interface vlan <i>vlan interface number</i> Example: Device(config)# ip radius source-interface vlan 80	Specifies that the RADIUS packets have the IP address of the indicated interface.
Step 4	radius-server host {<i>hostname</i> <i>ip-address</i>} test username <i>username</i> Example: Device(config)# radius-server host 172.120.39.46 test username user1	Specifies the host name or IP address of the remote RADIUS server. The test username <i>username</i> option enables automated testing of the RADIUS server connection. The specified <i>username</i> does not need to be a valid user name. The key option specifies an authentication and encryption key to use between the switch and the RADIUS server. To use multiple RADIUS servers, reenter this command for each server.
Step 5	radius-server key <i>string</i> Example: Device(config)# radius-server key rad123	Configures the authorization and encryption key used between the switch and the RADIUS daemon running on the RADIUS server.
Step 6	radius-server dead-criteria tries <i>num-tries</i> Example: Device(config)# radius-server dead-criteria tries 30	Specifies the number of unanswered sent messages to a RADIUS server before considering the server to be inactive. The range of <i>num-tries</i> is 1 to 100.
Step 7	end Example:	Returns to privileged EXEC mode.

	Command or Action	Purpose
	Device(config)# end	

Configuring the HTTP Server

To use web-based authentication, you must enable the HTTP server within the Device. You can enable the server for either HTTP or HTTPS.



Note The Apple psuedo-browser will not open if you configure only the **ip http secure-server** command. You should also configure the **ip http server** command.

Follow these steps to enable the server for either HTTP or HTTPS:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip http server**
4. **ip http secure-server**
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	ip http server Example: Device(config)# ip http server	Enables the HTTP server. The web-based authentication feature uses the HTTP server to communicate with the hosts for user authentication.
Step 4	ip http secure-server Example:	Enables HTTPS. You can configure custom authentication proxy web pages or specify a redirection URL for successful login.

	Command or Action	Purpose
	Device(config)# ip http secure-server	Note To ensure secure authentication when you enter the ip http secure-server command, the login page is always in HTTPS (secure HTTP) even if the user sends an HTTP request.
Step 5	end Example: Device(config)# end	Returns to privileged EXEC mode.

Customizing the Authentication Proxy Web Pages

You can configure web authentication to display four substitute HTML pages to the user in place of the Device default HTML pages during web-based authentication.

Follow these steps to specify the use of your custom authentication proxy web pages:

Before you begin

Store your custom HTML files on the Device flash memory.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip admission proxy http login page file** *device:login-filename*
4. **ip admission proxy http success page file** *device:success-filename*
5. **ip admission proxy http failure page file** *device:fail-filename*
6. **ip admission proxy http login expired page file** *device:expired-filename*
7. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.

	Command or Action	Purpose
Step 3	ip admission proxy http login page file <i>device:login-filename</i> Example: <pre>Device(config)# ip admission proxy http login page file disk1:login.htm</pre>	Specifies the location in the Device memory file system of the custom HTML file to use in place of the default login page. The <i>device:</i> is flash memory.
Step 4	ip admission proxy http success page file <i>device:success-filename</i> Example: <pre>Device(config)# ip admission proxy http success page file disk1:success.htm</pre>	Specifies the location of the custom HTML file to use in place of the default login success page.
Step 5	ip admission proxy http failure page file <i>device:fail-filename</i> Example: <pre>Device(config)# ip admission proxy http fail page file disk1:fail.htm</pre>	Specifies the location of the custom HTML file to use in place of the default login failure page.
Step 6	ip admission proxy http login expired page file <i>device:expired-filename</i> Example: <pre>Device(config)# ip admission proxy http login expired page file disk1:expired.htm</pre>	Specifies the location of the custom HTML file to use in place of the default login expired page.
Step 7	end Example: <pre>Device(config)# end</pre>	Returns to privileged EXEC mode.

Specifying a Redirection URL for Successful Login

Follow these steps to specify a URL to which the user is redirected after authentication, effectively replacing the internal Success HTML page:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip admission proxy http success redirect *url-string***

4. end

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	ip admission proxy http success redirect <i>url-string</i> Example: Device(config)# ip admission proxy http success redirect www.example.com	Specifies a URL for redirection of the user in place of the default login success page.
Step 4	end Example: Device(config)# end	Returns to privileged EXEC mode.

Configuring the Web-Based Authentication Parameters

Follow these steps to configure the maximum number of failed login attempts before the client is placed in a watch list for a waiting period:

SUMMARY STEPS

1. enable
2. configure terminal
3. ip admission max-login-attempts *number*
4. end
5. show running-config
6. copy running-config startup-config

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable	Enables privileged EXEC mode.

	Command or Action	Purpose
	Example: Device> <code>enable</code>	<ul style="list-style-type: none"> Enter your password if prompted.
Step 2	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode.
Step 3	ip admission max-login-attempts <i>number</i> Example: Device(config)# <code>ip admission max-login-attempts 10</code>	Sets the maximum number of failed login attempts. The range is 1 to 2147483647 attempts. The default is 5.
Step 4	end Example: Device(config)# <code>end</code>	Returns to privileged EXEC mode.
Step 5	show running-config Example: Device# <code>show running-config</code>	Verifies your entries.
Step 6	copy running-config startup-config Example: Device# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Configuring a Web Authentication Local Banner

For the equivalent Session Aware Networking configuration example for this feature, see the section "Configuring a Parameter Map for Web-Based Authentication" in the chapter, "Configuring Identity Control Policies" of the book, "*Session Aware Networking Configuration Guide, Cisco IOS XE Release 3SE (Catalyst 3850 Switches)*."

Beginning in privileged EXEC mode, follow these steps to configure a local banner on a switch that has web authentication configured.

SUMMARY STEPS

1. `configure terminal`

2. **ip auth-proxy auth-proxy-banner http** [*banner-text* | *file-path*]
3. **end**
4. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 2	ip auth-proxy auth-proxy-banner http [<i>banner-text</i> <i>file-path</i>] Example: Device(config)# aaa ip auth-proxy auth-proxy-banner C My Switch C	Enables the local banner. (Optional) Create a custom banner by entering <i>C banner-text C</i> , where <i>C</i> is a delimiting character or a file-path indicates a file (for example, a logo or text file) that appears in the banner.
Step 3	end Example: Device(config)# end	Returns to privileged EXEC mode.
Step 4	copy running-config startup-config Example: Device(config)# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Configuring Web-Based Authentication without SVI

You configure the web-based authentication without SVI feature to redirect the HTML login page to the client without creating an IP address in the routing table. These steps are optional.

You configure the web-based authentication without SVI feature to redirect the HTML login page to the client. This is done without creating an IP address in the SVI interface which then would be applied to the WebAuth enabled interface. These steps are optional.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **parameter-map type webauth global**
4. **l2-webauth-enabled**
5. **end**
6. **show running-config**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	parameter-map type webauth global Example: Device (config)# parameter-map type webauth global	Creates a parameter map and enters parameter-map webauth configuration mode. The specific configuration commands supported for a global parameter map defined with the global keyword differ from the commands supported for a named parameter map defined with the parameter-map-name argument.
Step 4	l2-webauth-enabled Example: Device (config-params-parameter-map)# l2-webauth-enabled	Enables the web-based authentication without SVI feature
Step 5	end Example: Device(config)# end	Returns to privileged EXEC mode.
Step 6	show running-config Example: Device# show running-config	Verifies your entries.
Step 7	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Configuring Web-Based Authentication with VRF Aware

You configure the web-based authentication with VRF aware to redirect the HTML login page to the client. These steps are optional.

SUMMARY STEPS

1. enable
2. configure terminal
3. parameter-map type webauth global
4. webauth-vrf-aware
5. end
6. show running-config
7. copy running-config startup-config

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	parameter-map type webauth global Example: Device (config)# parameter-map type webauth global	Creates a parameter map and enters parameter-map webauth configuration mode. The specific configuration commands supported for a global parameter map defined with the global keyword differ from the commands supported for a named parameter map defined with the parameter-map-name argument.
Step 4	webauth-vrf-aware Example: Device (config-params-parameter-map)# webauth-vrf-aware	Enables the web-based authentication VRF aware feature on SVI.
Step 5	end Example: Device (config)# end	Returns to privileged EXEC mode.
Step 6	show running-config Example: Device# show running-config	Verifies your entries.

	Command or Action	Purpose
Step 7	copy running-config startup-config Example: Device# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Removing Web-Based Authentication Cache Entries

Follow these steps to remove web-based authentication cache entries:

SUMMARY STEPS

1. `enable`
2. `clear ip auth-proxy cache {* | host ip address}`
3. `clear ip admission cache {* | host ip address}`

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> <code>enable</code>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	clear ip auth-proxy cache {* host ip address} Example: Device# <code>clear ip auth-proxy cache 192.168.4.5</code>	Delete authentication proxy entries. Use an asterisk to delete all cache entries. Enter a specific IP address to delete the entry for a single host.
Step 3	clear ip admission cache {* host ip address} Example: Device# <code>clear ip admission cache 192.168.4.5</code>	Delete authentication proxy entries. Use an asterisk to delete all cache entries. Enter a specific IP address to delete the entry for a single host.

Monitoring Web-Based Authentication Status

Use the commands in this topic to display the web-based authentication settings for all interfaces or for specific ports.

Table 151: Privileged EXEC show Commands

Command	Purpose
show authentication sessions method webauth	Displays the web-based authentication settings for all interfaces for fastethernet, gigabitethernet, or tengigabitethernet
show wireless client mac-address <i>a.a.a</i> detail	Displays the session specific wireless information and wireless states.
show authentication sessions interface <i>type slot/port</i>[details]	Displays the web-based authentication settings for the specified interface for fastethernet, gigabitethernet, or tengigabitethernet. In Session Aware Networking mode, use the show access-session interface command.

Displaying Web-Based Authentication Status

Perform this task to display the web-based authentication settings for all interfaces or for specific ports:

SUMMARY STEPS

1. **show authentication sessions {*interfacetype/ slot*}**

DETAILED STEPS

	Command or Action	Purpose
Step 1	show authentication sessions {<i>interfacetype/ slot</i>} Example: This example shows how to view only the global web-based authentication status: <pre>Switch# show authentication sessions</pre> Example: This example shows how to view the web-based authentication settings for gigabit interface 3/27: <pre>Switch# show authentication sessions interface gigabitethernet 3/27</pre>	Displays the web-based authentication settings. type = fastethernet, gigabitethernet, or tengigabitethernet (Optional) Use the interface keyword to display the web-based authentication settings for a specific interface

Monitoring HTTP Authentication Proxy

Perform the following task to troubleshoot your HTTP authentication proxy configuration:

SUMMARY STEPS

1. **enable**
2. **debug ip auth-proxy detailed**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	debug ip auth-proxy detailed Example: Device# debug ip auth-proxy detailed	Displays the authentication proxy configuration information on the device.

Verifying HTTPS Authentication Proxy

To verify your HTTPS authentication proxy configuration, perform the following optional steps:

SUMMARY STEPS

1. enable
2. show ip auth-proxy configuration
3. show ip auth-proxy cache
4. show ip http server secure status

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	show ip auth-proxy configuration Example: Device# show ip auth-proxy configuration	Displays the current authentication proxy configuration.
Step 3	show ip auth-proxy cache Example: Device# show ip auth-proxy cache	Displays the list of user authentication entries. The authentication proxy cache lists the host IP address, the source port number, the timeout value for the authentication proxy, and the state of the connection. If the authentication proxy state is HTTP_ESTAB, the user authentication was successful.
Step 4	show ip http server secure status Example:	Displays HTTPS status.

Command or Action	Purpose
Device# show ip http server secure status	

Configuration Examples for Web-Based Authentication

Example: Configuring the Authentication Rule and Interfaces

This example shows how to enable web-based authentication on Fast Ethernet port 5/1 :

```
Device(config)# ip admission name webauth1 proxy http
Device(config)# interface fastethernet 5/1
Device(config-if)# ip admission webauth1
Device(config-if)# exit
Device(config)# ip device tracking
```

This example shows how to verify the configuration:

```
Device# show ip admission status
IP admission status:
  Enabled interfaces           0
  Total sessions              0
  Init sessions               0      Max init sessions allowed    100
    Limit reached             0      Hi watermark                 0
  TCP half-open connections   0      Hi watermark                 0
  TCP new connections         0      Hi watermark                 0
  TCP half-open + new         0      Hi watermark                 0
  HTTPD1 Contexts            0      Hi watermark                 0

Parameter Map: Global
  Custom Pages
    Custom pages not configured
  Banner
    Banner not configured
```

Example: AAA Configuration

```
aaa new-model
aaa authentication login default group tacacs group radius
! Set up the aaa new model to use the authentication proxy.
aaa authorization auth-proxy default group tacacs group radius
! Define the AAA servers used by the router.
aaa accounting auth-proxy default start-stop group tacacs+
! Set up authentication proxy with accounting.
tacacs-server host 172.31.54.143
tacacs-server key cisco
radius-server host 172.31.54.143
radius-server key cisco
```

Example: HTTP Server Configuration

```
! Enable the HTTP server on the router.
```

```

ip http server
! Set the HTTP server authentication method to AAA.
ip http authentication aaa
! Define standard access list 61 to deny any host.
access-list 61 deny any
! Use ACL 61 to deny connections from any host to the HTTP server.
ip http access-class 61

```

Example: Customizing the Authentication Proxy Web Pages

This example shows how to configure custom authentication proxy web pages:

```

Device(config)# ip admission proxy http login page file flash:login.htm
Device(config)# ip admission proxy http success page file flash:success.htm
Device(config)# ip admission proxy http fail page file flash:fail.htm
Device(config)# ip admission proxy http login expired page flash:expired.htm

```

This example shows how to verify the configuration of a custom authentication proxy web pages:

```

Device# show ip admission configuration
Authentication proxy webpage
Login page : flash:login.htm
Success page : flash:success.htm
Fail Page : flash:fail.htm
Login expired Page : flash:expired.htm

Authentication global cache time is 60 minutes
Authentication global absolute time is 0 minutes
Authentication global init state time is 2 minutes
Authentication Proxy Session ratelimit is 100
Authentication Proxy Watch-list is disabled
Authentication Proxy Auditing is disabled
Max Login attempts per user is 5

```

Example: Specifying a Redirection URL for Successful Login

Configuring redirection URL for successful login

```

Device(config)# ip admission proxy http success redirect www.cisco.com

```

Verifying redirection URL for Successful Login

This example shows how to configure a redirection URL for successful login:

```

Device# show ip admission status
Enabled interfaces          0
Total sessions             0
Init sessions              0   Max init sessions allowed   100
  Limit reached            0   Hi watermark                0
TCP half-open connections  0   Hi watermark                0
TCP new connections        0   Hi watermark                0
TCP half-open + new       0   Hi watermark                0
HTTPD1 Contexts           0   Hi watermark                0

```

```

Parameter Map: Global
  Custom Pages
    Custom pages not configured
  Banner
    Banner not configured

```

Additional References for Web-Based Authentication

Related Documents

Related Topic	Document Title
IBNS commands	Cisco IOS Identity-Based Networking Services Command Reference
Wired guest access	<i>Wired Guest Access</i> chapter

Technical Assistance

Description	Link
<p>The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies.</p> <p>To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds.</p> <p>Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.</p>	http://www.cisco.com/support

Feature Information for Web-Based Authentication

Release	Feature Information
Cisco IOS Release 15.0(2)EX1	This feature is introduced.



CHAPTER 76

Auto Identity

- [Auto Identity, on page 1573](#)

Auto Identity

The Auto Identity feature provides a set of built-in policies at global configuration and interface configuration modes. This feature is available only in Class-Based Policy Language (CPL) control policy-equivalent new-style mode. To convert all the relevant authentication commands to their CPL control policy-equivalents, use the **authentication convert-to new-style** command.

This module describes the feature and explains how to configure it.

Information About Auto Identity

Auto Identity Overview

The Cisco Identity-Based Networking Services (IBNS) solution provides a policy and identity-based framework in which edge devices can deliver flexible and scalable services to subscribers. IBNS allows the concurrent operation of IEEE 802.1x (dot1x), MAC authentication bypass (MAB), and web authentication methods, making it possible to invoke multiple authentication methods in parallel, on a single subscriber session. These authentication methods, dot1x, authentication, authorization, and accounting (AAA), and RADIUS are available in global configuration and interface configuration modes.

The Auto Identity feature uses the Cisco Common Classification Policy Language-based configuration that significantly reduces the number of commands used to configure both authentication methods and interface-level commands. The Auto Identity feature provides a set of built-in policies that are based on policy maps, class maps, parameter maps, and interface templates.

In global configuration mode, the **source template AI_GLOBAL_CONFIG_TEMPLATE** command enables the Auto Identity feature. In interface configuration mode, configure the **AI_MONITOR_MODE**, **AI_LOW_IMPACT_MODE**, or **AI_CLOSED_MODE** interface templates to enable the feature on interfaces.

You can configure multiple templates; however, you must bind multiple templates together using the **merge** command. If you do not bind the templates, the last configured template is used. While binding templates, if the same command is repeated in two templates with different arguments, the last configured command is used.



Note You can also enable user-defined templates that are configured using the **template name** command in global configuration mode .

Use the **show template interface** or **show template global** commands to display information about built-in templates. Built-in templates can be edited. Built-in template information is displayed in the output of the **show running-config** command, if the template is edited. If you delete an edited built-in template, the built-in template reverts to the default and is not deleted from the configuration. However, if you delete a user-defined template, it is deleted from the configuration.



Note Before you delete a template, ensure that it is not attached to a device.

Auto Identity Global Template

To enable the global template, configure the **source template template-name** command.



Note You must configure the RADIUS server commands, because these are not automatically configured when the global template is enabled.

The following example shows how to enable the global template:

```
Switch(config)# source template AI_GLOBAL_CONFIG_TEMPLATE
Switch(config)# radius server ISE
Switch(config-radius-server)# address ipv4 172.20.254.4 auth-port 1645 acct-port 1646
Switch(config-radius-server)# key cisco
Switch(config-radius-server)# end
```

The AI_GLOBAL_CONFIG_TEMPLATE automatically configures the following commands:

```
dot1x system-auth-control
aaa new-model
aaa authentication dot1x default group radius
aaa authorization network default group radius
aaa authorization auth-proxy default group radius
aaa accounting identity default start-stop group radius
aaa accounting system default start-stop group radius
radius-server attribute 6 on-for-login-auth
radius-server attribute 6 support-multiple
radius-server attribute 6 voice 1
radius-server attribute 8 include-in-access-req
radius-server attribute 25 access-request include
```

Auto Identity Interface Templates

The following interface templates are available in the Auto Identity feature:

- AI_MONITOR_MODE—Passively monitors sessions that have authentication in open mode.
- AI_LOW_IMPACT_MODE—Similar to monitor mode, but with a configured static policy such as a port access control list (PACL).

- **AI_CLOSED_MODE**—Secure mode in which data traffic is not allowed into the network, until authentication is complete. This mode is the default.



Note Multi-auth host mode is not supported with the LAN Lite license.

The following commands are inbuilt in the **AI_MONITOR_MODE**:

```
switchport mode access
access-session port-control auto
access-session host-mode multi-auth
dot1x pae authenticator
mab
service-policy type control subscriber AI_DOT1X_MAB_POLICIES
```

The following commands are inbuilt in the **AI_LOW_IMPACT_MODE**:

```
switchport mode access
access-session port-control auto
access-session host-mode multi-auth
dot1x pae authenticator
mab
ip access-group AI_PORT_ACL in
service-policy type control subscriber AI_DOT1X_MAB_POLICIES
```

The following commands are inbuilt in the **AI_CLOSED_MODE**:

```
switchport mode access
access-session closed
access-session port-control auto
access-session host-mode multi-auth
dot1x pae authenticator
mab
service-policy type control subscriber AI_DOT1X_MAB_POLICIES
```

Auto Identity Built-in Policies

The following five built-in policies are available in the Auto Identity feature:

- **AI_DOT1X_MAB_AUTH**—Enables flexible authentication with dot1x, and then MAC Address Bypass (MAB).
- **AI_DOT1X_MAB_POLICIES**—Enables flexible authentication with dot1x, and then MAB. Applies critical VLAN in case the Authentication, Authorization, and Accounting (AAA) server is not reachable.
- **AI_DOT1X_MAB_WEBAUTH**—Enables flexible authentication with dot1x, MAB, and then web authentication.
- **AI_NEXTGEN_AUTHBYBASS**—Skips authentication if an IP phone device is detected. Enables the **device classifier** command in global configuration mode and the **voice-vlan** command in interface configuration mode to detect the device. This is a reference policy map, and users can copy the contents of this policy map to other policy maps.
- **AI_STANDALONE_WEBAUTH**—Defines standalone web authentication.

Auto Identity Class Maps Templates

The following built-in class maps are supported by the Auto Identity feature:

- **AI_NRH**—Specifies that the nonresponsive host (NRH) authentication method is enabled.
- **AI_WEBAUTH_METHOD**—Specifies that the web authentication method is enabled.
- **AI_WEBAUTH_FAILED**—Specifies that the web authentication method failed to authenticate.
- **AI_WEBAUTH_NO_RESP**—Specifies that the web authentication client failed to respond.
- **AI_DOT1X_METHOD**—Specifies that the dot1x method is enabled.
- **AI_DOT1X_FAILED**—Specifies that the dot1x method failed to authenticate.
- **AI_DOT1X_NO_RESP**—Specifies that the dot1x client failed to respond.
- **AI_DOT1X_TIMEOUT**—Specifies that the dot1x client stopped responding after the initial acknowledge (ACK) request.
- **AI_MAB_METHOD**—Specifies that the MAC Authentication Bypass (MAB) method is enabled.
- **AI_MAB_FAILED**—Specifies that the MAB method failed to authenticate.
- **AI_AAA_SVR_DOWN_AUTHD_HOST**—Specifies that the Authentication, Authorization, and Accounting (AAA) server is down, and the client is in authorized state.
- **AI_AAA_SVR_DOWN_UNAUTHD_HOST**—Specifies that the AAA server is down, and the client is in authorized state.
- **AI_IN_CRITICAL_AUTH**—Specifies that the critical authentication service template is applied.
- **AI_NOT_IN_CRITICAL_AUTH**—Specifies that the critical authentication service template is not applied.
- **AI_METHOD_DOT1X_DEVICE_PHONE**—Specifies that the method is dot1x and the device type is IP phone.
- **AI_DEVICE_PHONE**—Specifies that the device type is IP phone.

Auto Identity Parameter Maps

The following built-in parameter map templates are supported by the Auto Identity feature:

- **AI_NRH_PMAP**—Starts nonresponsive host (NRH) authentication.
- **AI_WEBAUTH_PMAP**—Starts web authentication.

Auto Identity Service Templates

Service templates are available inside built-in policy maps. The following built-in service templates are supported by the Auto Identity feature:

- **AI_INACTIVE_TIMER**—Template to start the inactivity timer.
- **AI_CRITICAL_ACL**—Dummy template; users can configure this template as per their requirements.

How to Configure Auto Identity

Configuring Auto Identity Globally

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **sourcetemplate** {**AI_GLOBAL_CONFIG_TEMPLATE** | *template-name*}
4. **aaa new-model**
5. **radius server** *name*

6. **address ipv4** {*hostname* | *ipv4-address*}
7. **key ipv4** {**0** *string* | **7** *string*} *string*
8. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	sourcetemplate { AI_GLOBAL_CONFIG_TEMPLATE <i>template-name</i> } Example: Switch(config)# source template AI_GLOBAL_CONFIG_TEMPLATE	Configures an auto identity template. <ul style="list-style-type: none"> • AI_GLOBAL_CONFIG_TEMPLATE is a built-in template. • <i>template-name</i> is a user-defined template.
Step 4	aaa new-model Example: Switch(config)# aaa new-model	Enables the authentication, authorization, and accounting (AAA) access control mode.
Step 5	radius server <i>name</i> Example: Switch(config)# radius server ISE	Specifies the name for the RADIUS server configuration for Protected Access Credential (PAC) provisioning and enters RADIUS server configuration mode.
Step 6	address ipv4 { <i>hostname</i> <i>ipv4-address</i> } Example: Switch(config-radius-server)# address ipv4 10.1.1.1	Configures the IPv4 address for the RADIUS server accounting and authentication parameters. <p>Note This command is not a part of the global template, and you must configure it.</p>
Step 7	key ipv4 { 0 <i>string</i> 7 <i>string</i> } <i>string</i> Example: Switch(config-radius-server)# key ipv4 cisco	Specifies the authentication and encryption key for all RADIUS communications between the device and the RADIUS server. <p>Note This command is not a part of the global template, and you must configure it.</p>
Step 8	end Example: Switch(config-radius-server)# end	Exits RADIUS server configuration mode and returns to privileged EXEC mode.

Configuring Auto Identity at an Interface Level

When you configure two interface templates, you must configure the **merge** keyword. If you do not, the last configured template is used.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *type number*
4. **source template** {**AI_CLOSED_MODE** | **AI_LOW_IMPACT_MODE** | **AI_MONITOR_MODE** | *template-name*} [**merge**]
5. **source template** {**AI_CLOSED_MODE** | **AI_LOW_IMPACT_MODE** | **AI_MONITOR_MODE** | *template-name*} [**merge**]
6. **switchport access vlan** *vlan-id*
7. **switchport voice vlan** *vlan-id*
8. Repeat Steps 4, 6, and 7 on all interfaces that must have the Auto Identity feature configured.
9. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. <ul style="list-style-type: none">• Enter your password if prompted.
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	interface <i>type number</i> Example: Switch(config)# interface gigabitethernet 1/0/1	Configures an interface and enters interface configuration mode.
Step 4	source template { AI_CLOSED_MODE AI_LOW_IMPACT_MODE AI_MONITOR_MODE <i>template-name</i> } [merge] Example: Switch(config-if)# source template AI_CLOSED_MODE	Configures a source template for the interface.
Step 5	source template { AI_CLOSED_MODE AI_LOW_IMPACT_MODE AI_MONITOR_MODE <i>template-name</i> } [merge] Example: Switch(config-if)# source template AI_MONITOR_MODE merge	(Optional) Configures a source template for the interface and merges this template with the previously configured template <ul style="list-style-type: none">• When you configure two templates, if you do not configure the merge keyword, the last configured template is used.

	Command or Action	Purpose
Step 6	switchport access vlan <i>vlan-id</i> Example: Switch(config-if)# switchport access vlan 100	Sets the VLAN when the interface is in access mode.
Step 7	switchport voice vlan <i>vlan-id</i> Example: Switch(config-if)# switchport voice vlan 101	Configures a voice VLAN on a multiple VLAN access port.
Step 8	Repeat Steps 4, 6, and 7 on all interfaces that must have the Auto Identity feature configured.	—
Step 9	end Example: Switch(config-if)# end	Exits interface configuration mode and returns to privileged EXEC mode.

Configuration Examples for Auto Identity

Example: Configuring Auto Identity Globally

```
Switch> enable
Switch# configure terminal
Switch(config)# source template AI_GLOBAL_CONFIG_TEMPLATE
Switch(config)# aaa new-model
Switch(config)# radius server ISE
Switch(config-radius-server)# address ipv4 10.1.1.1
Switch(config-radius-server)# key ipv4 cisco
Switch(config-radius-server)# end
```

Example: Configuring Auto Identity at an Interface Level

```
Switch> enable
Switch# configure terminal
Switch(config)# interface gigabitethernet 1/0/1
Switch(config-if)# source template AI_CLOSED_MODE
Switch(config-if)# source template AI_MONITOR_MODE merge
Switch(config-if)# switchport access vlan 100
Switch(config-if)# switchport voice vlan 101
Switch(config-if)# end
```

Verifying Auto Identity

Step 1 enable
Example:

```
Switch> enable
```

Enables Privileged EXEC mode.

- Enter your password if prompted.

Step 2 show template interface source built-in all

Displays all the configured built-in interface templates.

Example:

```
Switch# show template interface source built-in all

Template Name       : AI_CLOSED_MODE
Modified           : No
Template Definition :
dot1x pae authenticator
switchport mode access
mab
access-session closed
access-session port-control auto
service-policy type control subscriber AI_DOT1X_MAB_POLICIES
!
Template Name       : AI_LOW_IMPACT_MODE
Modified           : No
Template Definition :
dot1x pae authenticator
switchport mode access
mab
access-session port-control auto
service-policy type control subscriber AI_DOT1X_MAB_POLICIES
ip access-group AI_PORT_ACL in
!
Template Name       : AI_MONITOR_MODE
Modified           : No
Template Definition :
dot1x pae authenticator
switchport mode access
mab
access-session port-control auto
service-policy type control subscriber AI_DOT1X_MAB_POLICIES
!
```

Step 3 show template global source built-in all

Displays all the configured global built-in templates.

Example:

```
Switch# show template global source built-in all

Global Template Name       : AI_GLOBAL_CONFIG_TEMPLATE
Modified                   : No
Global Template Definition : global
dot1x system-auth-control
aaa new-model
aaa authentication dot1x default group radius
aaa authorization network default group radius
aaa authorization auth-proxy default group radius
aaa accounting identity default start-stop group radius
aaa accounting system default start-stop group radius
radius-server attribute 6 on-for-login-auth
radius-server attribute 6 support-multiple
radius-server attribute 6 voice 1
```

```
radius-server attribute 8 include-in-access-req
radius-server attribute 25 access-request include
!
```

Step 4 **show derived-config | include aaa | radius-server**

Displays the composite results of all the configuration commands that apply to an interface, including commands that come from sources such as static templates, dynamic templates, dialer interfaces, and authentication, authorization, and accounting (AAA) per-user attributes.

Example:

```
Switch# show derived-config | inc aaa | radius-server

aaa new-model
aaa authentication dot1x default group radius
aaa authorization network default group radius
aaa authorization auth-proxy default group radius
aaa accounting identity default start-stop group radius
aaa accounting system default start-stop group radius
aaa session-id common
radius-server attribute 6 on-for-login-auth
radius-server attribute 6 support-multiple
radius-server attribute 6 voice 1
radius-server attribute 8 include-in-access-req
radius-server attribute 25 access-request include
radius-server host 10.25.18.42 key cisco
```

Step 5 **show derived-config | interface type-number**

Displays the composite results of all configuration for an interface.

Example:

```
Switch# show derived-config | interface gigabitethernet2/0/6

Building configuration...

Derived configuration : 267 bytes
!
interface GigabitEthernet2/0/6
 switchport mode access
 switchport voice vlan 100
 access-session closed
 access-session port-control auto
 mab
 dot1x pae authenticator
 spanning-tree portfast edge
 service-policy type control subscriber AI_DOT1X_MAB_POLICIES
end
```

Step 6 **show access-session | interface interface-type-number details**

Displays the policies applied to an interface.

Example:

```
Switch# show access-session interface gigabitethernet2/0/6 details

Interface: GigabitEthernet2/0/6
```

```

MAC Address: c025.5c43.be00
IPv6 Address: Unknown
IPv4 Address: Unknown
  User-Name: CP-9971-SEPC0255C43BE00
  Device-type: Cisco-IP-Phone-9971
    Status: Authorized
    Domain: VOICE
  Oper host mode: multi-auth
  Oper control dir: both
  Session timeout: N/A
Common Session ID: 091A1C5B00000017002003EE
Acct Session ID: 0x00000005
  Handle: 0xBB00000B
  Current Policy: AI_DOT1X_MAB_POLICIES

```

Local Policies:

Server Policies:

```

  Vlan Group: Vlan: 100
Security Policy: Must Not Secure
Security Status: Link Unsecure

```

Method status list:

```

Method      State
dot1x      Authc Success

```

Step 7 **show running-config interface *type-number***

Displays the contents of the current running configuration file or the configuration for an interface.

Example:

```

Switch# show running-config interface gigabitethernet2/0/6

Building configuration...

Current configuration : 214 bytes
!
interface GigabitEthernet2/0/6
 switchport mode access
 switchport voice vlan 100
 access-session port-control auto
 spanning-tree portfast edge
 service-policy type control subscriber AI_NEXTGEN_AUTHBYPASS
end

```

Step 8 **show lldp neighbor**

Displays information about one or all neighboring devices discovered using the Link Layer Discovery Protocol (LLDP).

Example:

```

Switch# show lldp neighbor

Capability codes:
 (R) Router, (B) Bridge, (T) Telephone, (C) DOCSIS Cable Device
 (W) WLAN Access Point, (P) Repeater, (S) Station, (O) Other

Device ID           Local Intf      Hold-time  Capability  Port ID
SEPC0255C43BE00    Gi2/0/6        180        B,T         C0255C43BE00:P1

```


Total entries displayed: 1

Feature Information for Auto Identity

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Table 152: Feature Information for Auto Identity

Feature Name	Releases	Feature Information
Auto Identity	Cisco IOS Release 15.2(4)E	<p>The Auto Identity feature provides a set of built-in policies at the global configuration and interface configuration modes. This feature is available only in the Class-Based Policy Language (CPL) control policy-equivalent new-style mode.</p> <p>In Cisco IOS Release 15.2(4)E, this feature was implemented on Cisco Catalyst 2960–X Series Switches, Catalyst 3750–X Series Switches, and Cisco Catalyst 4500E Supervisor Engine 7-E.</p> <p>The following commands was introduced or modified: source-template.</p>



CHAPTER 77

Configuring Port-Based Traffic Control

- [Overview of Port-Based Traffic Control](#) , on page 1586
- [Finding Feature Information](#), on page 1586
- [Information About Storm Control](#), on page 1586
- [How to Configure Storm Control](#), on page 1588
- [Finding Feature Information](#), on page 1592
- [Information About Protected Ports](#), on page 1593
- [How to Configure Protected Ports](#), on page 1593
- [Monitoring Protected Ports](#), on page 1595
- [Where to Go Next](#), on page 1595
- [Additional References](#), on page 1595
- [Feature Information](#), on page 1596
- [Finding Feature Information](#), on page 1596
- [Information About Port Blocking](#), on page 1596
- [How to Configure Port Blocking](#), on page 1597
- [Monitoring Port Blocking](#), on page 1598
- [Where to Go Next](#), on page 1599
- [Additional References](#), on page 1599
- [Feature Information](#), on page 1600
- [Prerequisites for Port Security](#), on page 1600
- [Restrictions for Port Security](#), on page 1600
- [Information About Port Security](#), on page 1600
- [How to Configure Port Security](#), on page 1605
- [Configuration Examples for Port Security](#), on page 1612
- [Additional References](#), on page 1612
- [Finding Feature Information](#), on page 1613
- [Information About Protocol Storm Protection](#), on page 1613
- [How to Configure Protocol Storm Protection](#), on page 1614
- [Monitoring Protocol Storm Protection](#), on page 1615
- [Additional References](#), on page 1615

Overview of Port-Based Traffic Control

Port-based traffic control is a set of Layer 2 features on the Cisco Catalyst switches used to filter or block packets at the port level in response to specific traffic conditions. The following port-based traffic control features are supported in the Cisco IOS Release for which this guide is written:

- Storm Control
- Protected Ports
- Port Blocking
- Port Security
- Protocol Storm Protection

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to <http://www.cisco.com/go/cfn>. An account on Cisco.com is not required.

Information About Storm Control

Storm Control

Storm control prevents traffic on a LAN from being disrupted by a broadcast, multicast, or unicast storm on one of the physical interfaces. A LAN storm occurs when packets flood the LAN, creating excessive traffic and degrading network performance. Errors in the protocol-stack implementation, mistakes in network configurations, or users issuing a denial-of-service attack can cause a storm.

Storm control (or traffic suppression) monitors packets passing from an interface to the switching bus and determines if the packet is unicast, multicast, or broadcast. The switch counts the number of packets of a specified type received within the 1-second time interval and compares the measurement with a predefined suppression-level threshold.

How Traffic Activity is Measured

Storm control uses one of these methods to measure traffic activity:

- Bandwidth as a percentage of the total available bandwidth of the port that can be used by the broadcast, multicast, or unicast traffic

- Traffic rate in packets per second at which broadcast, multicast, or unicast packets are received
- Traffic rate in bits per second at which broadcast, multicast, or unicast packets are received
- Traffic rate in packets per second and for small frames. This feature is enabled globally. The threshold for small frames is configured for each interface.

With each method, the port blocks traffic when the rising threshold is reached. The port remains blocked until the traffic rate drops below the falling threshold (if one is specified) and then resumes normal forwarding. If the falling suppression level is not specified, the switch blocks all traffic until the traffic rate drops below the rising suppression level. In general, the higher the level, the less effective the protection against broadcast storms.

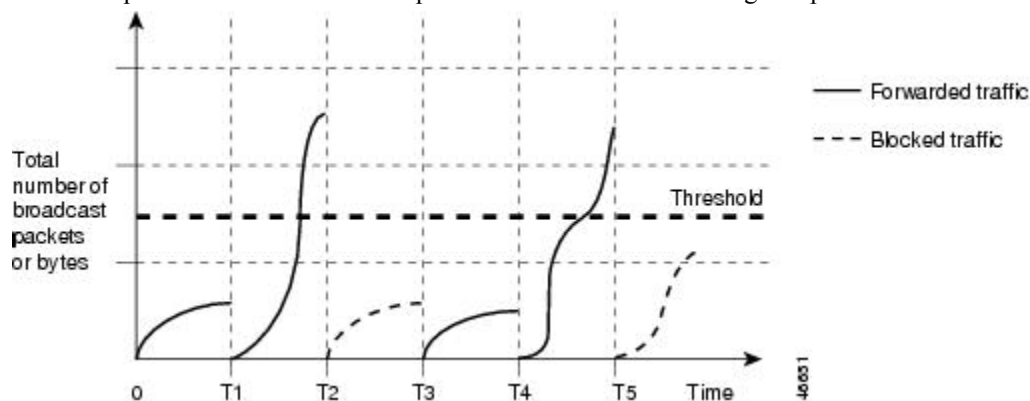


Note When the storm control threshold for multicast traffic is reached, all multicast traffic except control traffic, such as bridge protocol data unit (BDPU) and Cisco Discovery Protocol (CDP) frames, are blocked. However, the switch does not differentiate between routing updates, such as OSPF, and regular multicast data traffic, so both types of traffic are blocked.

Traffic Patterns

Figure 122: Broadcast Storm Control Example

This example shows broadcast traffic patterns on an interface over a given period of time.



Broadcast traffic being forwarded exceeded the configured threshold between time intervals T1 and T2 and between T4 and T5. When the amount of specified traffic exceeds the threshold, all traffic of that kind is dropped for the next time period. Therefore, broadcast traffic is blocked during the intervals following T2 and T5. At the next time interval (for example, T3), if broadcast traffic does not exceed the threshold, it is again forwarded.

The combination of the storm-control suppression level and the 1-second time interval controls the way the storm control algorithm works. A higher threshold allows more packets to pass through. A threshold value of 100 percent means that no limit is placed on the traffic. A value of 0.0 means that all broadcast, multicast, or unicast traffic on that port is blocked.



Note Because packets do not arrive at uniform intervals, the 1-second time interval during which traffic activity is measured can affect the behavior of storm control.

You use the **storm-control** interface configuration commands to set the threshold value for each traffic type.

How to Configure Storm Control

Configuring Storm Control and Threshold Levels

You configure storm control on a port and enter the threshold level that you want to be used for a particular type of traffic.

However, because of hardware limitations and the way in which packets of different sizes are counted, threshold percentages are approximations. Depending on the sizes of the packets making up the incoming traffic, the actual enforced threshold might differ from the configured level by several percentage points.



Note Storm control is supported on physical interfaces. You can also configure storm control on an EtherChannel. When storm control is configured on an EtherChannel, the storm control settings propagate to the EtherChannel physical interfaces.

Follow these steps to storm control and threshold levels:

Before you begin

Storm control is supported on physical interfaces. You can also configure storm control on an EtherChannel. When storm control is configured on an EtherChannel, the storm control settings propagate to the EtherChannel physical interfaces.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **storm-control** {**broadcast** | **multicast** | **unicast**} **level** {*level* [*level-low*] | **bps** *bps* [*bps-low*] | **pps** *pps* [*pps-low*]}
5. **storm-control action** {**shutdown** | **trap**}
6. **end**
7. **show storm-control** [*interface-id*] [**broadcast** | **multicast** | **unicast**]
8. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable	Enables privileged EXEC mode.

	Command or Action	Purpose
	<p>Example:</p> <pre>Device> enable</pre>	<ul style="list-style-type: none"> Enter your password if prompted.
Step 2	<p>configure terminal</p> <p>Example:</p> <pre>Device# configure terminal</pre>	Enters global configuration mode.
Step 3	<p>interface interface-id</p> <p>Example:</p> <pre>Device(config)# interface gigabitethernet1/0/1</pre>	Specifies the interface to be configured, and enter interface configuration mode.
Step 4	<p>storm-control {broadcast multicast unicast} level {level [level-low] bps bps [bps-low] pps pps [pps-low]}</p> <p>Example:</p> <pre>Device(config-if)# storm-control unicast level 87 65</pre>	<p>Configures broadcast, multicast, or unicast storm control. By default, storm control is disabled.</p> <p>The keywords have these meanings:</p> <ul style="list-style-type: none"> For <i>level</i>, specifies the rising threshold level for broadcast, multicast, or unicast traffic as a percentage (up to two decimal places) of the bandwidth. The port blocks traffic when the rising threshold is reached. The range is 0.00 to 100.00. (Optional) For <i>level-low</i>, specifies the falling threshold level as a percentage (up to two decimal places) of the bandwidth. This value must be less than or equal to the rising suppression value. The port forwards traffic when traffic drops below this level. If you do not configure a falling suppression level, it is set to the rising suppression level. The range is 0.00 to 100.00. <p>If you set the threshold to the maximum value (100 percent), no limit is placed on the traffic. If you set the threshold to 0.0, all broadcast, multicast, and unicast traffic on that port is blocked.</p> <ul style="list-style-type: none"> For bps bps, specifies the rising threshold level for broadcast, multicast, or unicast traffic in bits per second (up to one decimal place). The port blocks traffic when the rising threshold is reached. The range is 0.0 to 10000000000.0. (Optional) For <i>bps-low</i>, specifies the falling threshold level in bits per second (up to one decimal place). It can be less than or equal to the rising threshold level. The port forwards traffic when traffic drops below this level. The range is 0.0 to 10000000000.0.

	Command or Action	Purpose
		<ul style="list-style-type: none"> For pps pps, specifies the rising threshold level for broadcast, multicast, or unicast traffic in packets per second (up to one decimal place). The port blocks traffic when the rising threshold is reached. The range is 0.0 to 10000000000.0. (Optional) For pps-low, specifies the falling threshold level in packets per second (up to one decimal place). It can be less than or equal to the rising threshold level. The port forwards traffic when traffic drops below this level. The range is 0.0 to 10000000000.0. <p>For BPS and PPS settings, you can use metric suffixes such as k, m, and g for large number thresholds.</p>
Step 5	storm-control action {shutdown trap} Example: <pre>Device(config-if)# storm-control action trap</pre>	<p>Specifies the action to be taken when a storm is detected. The default is to filter out the traffic and not to send traps.</p> <ul style="list-style-type: none"> Select the shutdown keyword to error-disable the port during a storm. Select the trap keyword to generate an SNMP trap when a storm is detected.
Step 6	end Example: <pre>Device(config-if)# end</pre>	Returns to privileged EXEC mode.
Step 7	show storm-control [interface-id] [broadcast multicast unicast] Example: <pre>Device# show storm-control gigabitethernet1/0/1 unicast</pre>	Verifies the storm control suppression levels set on the interface for the specified traffic type. If you do not enter a traffic type, details for all traffic types (broadcast, multicast and unicast) are displayed.
Step 8	copy running-config startup-config Example: <pre>Device# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Configuring Small-Frame Arrival Rate

Incoming VLAN-tagged packets smaller than 67 bytes are considered small frames. They are forwarded by the switch, but they do not cause the switch storm-control counters to increment.

You globally enable the small-frame arrival feature on the switch and then configure the small-frame threshold for packets on each interface. Packets smaller than the minimum size and arriving at a specified rate (the threshold) are dropped since the port is error disabled.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **errdisable detect cause small-frame**
4. **errdisable recovery interval *interval***
5. **errdisable recovery cause small-frame**
6. **interface *interface-id***
7. **small-frame violation-rate *pps***
8. **end**
9. **show interfaces *interface-id***
10. **show running-config**
11. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	errdisable detect cause small-frame Example: Device(config)# errdisable detect cause small-frame	Enables the small-frame rate-arrival feature on the switch.
Step 4	errdisable recovery interval <i>interval</i> Example: Device(config)# errdisable recovery interval 60	(Optional) Specifies the time to recover from the specified error-disabled state.
Step 5	errdisable recovery cause small-frame Example: Device(config)# errdisable recovery cause small-frame	(Optional) Configures the recovery time for error-disabled ports to be automatically re-enabled after they are error disabled by the arrival of small frames Storm control is supported on physical interfaces. You can also configure storm control on an EtherChannel. When storm control is configured on an EtherChannel, the storm

	Command or Action	Purpose
		control settings propagate to the EtherChannel physical interfaces.
Step 6	interface <i>interface-id</i> Example: Device(config)# interface gigabitethernet1/0/2	Enters interface configuration mode, and specify the interface to be configured.
Step 7	small-frame violation-rate <i>pps</i> Example: Device(config-if)# small-frame violation rate 10000	Configures the threshold rate for the interface to drop incoming packets and error disable the port. The range is 1 to 10,000 packets per second (pps)
Step 8	end Example: Device(config)# end	Returns to privileged EXEC mode.
Step 9	show interfaces <i>interface-id</i> Example: Device# show interfaces gigabitethernet1/0/2	Verifies the configuration.
Step 10	show running-config Example: Device# show running-config	Verifies your entries.
Step 11	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to <http://www.cisco.com/go/cfn>. An account on Cisco.com is not required.

Information About Protected Ports

Protected Ports

Some applications require that no traffic be forwarded at Layer 2 between ports on the same switch so that one neighbor does not see the traffic generated by another neighbor. In such an environment, the use of protected ports ensures that there is no exchange of unicast, broadcast, or multicast traffic between these ports on the switch.

Protected ports have these features:

- A protected port does not forward any traffic (unicast, multicast, or broadcast) to any other port that is also a protected port. Data traffic cannot be forwarded between protected ports at Layer 2; only control traffic, such as PIM packets, is forwarded because these packets are processed by the CPU and forwarded in software. All data traffic passing between protected ports must be forwarded through a Layer 3 device.
- Forwarding behavior between a protected port and a nonprotected port proceeds as usual.

Because a switch stack represents a single logical switch, Layer 2 traffic is not forwarded between any protected ports in the switch stack, whether they are on the same or different switches in the stack.

Default Protected Port Configuration

The default is to have no protected ports defined.

Protected Ports Guidelines

You can configure protected ports on a physical interface (for example, Gigabit Ethernet port 1) or an EtherChannel group (for example, port-channel 5). When you enable protected ports for a port channel, it is enabled for all ports in the port-channel group.

How to Configure Protected Ports

Configuring a Protected Port

Before you begin

Protected ports are not pre-defined. This is the task to configure one.

SUMMARY STEPS

1. `enable`

2. **configure terminal**
3. **interface *interface-id***
4. **switchport protected**
5. **end**
6. **show interfaces *interface-id* switchport**
7. **show running-config**
8. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	interface <i>interface-id</i> Example: Device(config)# interface gigabitethernet 1/0/1	Specifies the interface to be configured, and enter interface configuration mode.
Step 4	switchport protected Example: Device(config-if)# switchport protected	Configures the interface to be a protected port.
Step 5	end Example: Device(config)# end	Returns to privileged EXEC mode.
Step 6	show interfaces <i>interface-id</i> switchport Example: Device# show interfaces gigabitethernet 1/0/1 switchport	Verifies your entries.

	Command or Action	Purpose
Step 7	show running-config Example: Device# <code>show running-config</code>	Verifies your entries.
Step 8	copy running-config startup-config Example: Device# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Monitoring Protected Ports

Table 153: Commands for Displaying Protected Port Settings

Command	Purpose
<code>show interfaces [interface-id] switchport</code>	Displays the administrative and operational status of all switching (nonrouting) ports or the specified port, including port blocking and port protection settings.

Where to Go Next

.

Additional References

MIBs

MIB	MIBs Link
All the supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
<p>The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies.</p> <p>To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds.</p> <p>Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.</p>	http://www.cisco.com/support

Feature Information

Release	Feature Information
Cisco IOS Release 15.0(2)EX1	This feature was introduced.

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to <http://www.cisco.com/go/cfn>. An account on Cisco.com is not required.

Information About Port Blocking

Port Blocking

By default, the switch floods packets with unknown destination MAC addresses out of all ports. If unknown unicast and multicast traffic is forwarded to a protected port, there could be security issues. To prevent unknown unicast or multicast traffic from being forwarded from one port to another, you can block a port (protected or nonprotected) from flooding unknown unicast or multicast packets to other ports.



Note With multicast traffic, the port blocking feature blocks only pure Layer 2 packets. Multicast packets that contain IPv4 or IPv6 information in the header are not blocked.

How to Configure Port Blocking

Blocking Flooded Traffic on an Interface

Before you begin

The interface can be a physical interface or an EtherChannel group. When you block multicast or unicast traffic for a port channel, it is blocked on all ports in the port-channel group.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **switchport block multicast**
5. **switchport block unicast**
6. **end**
7. **show interfaces** *interface-id* **switchport**
8. **show running-config**
9. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	interface <i>interface-id</i> Example: Device(config)# interface gigabitethernet 1/0/1	Specifies the interface to be configured, and enter interface configuration mode.
Step 4	switchport block multicast Example:	Blocks unknown multicast forwarding out of the port.

	Command or Action	Purpose
	Device(config-if)# switchport block multicast	
Step 5	switchport block unicast Example: Device(config-if)# switchport block unicast	Blocks unknown unicast forwarding out of the port.
Step 6	end Example: Device(config)# end	Returns to privileged EXEC mode.
Step 7	show interfaces interface-id switchport Example: Device# show interfaces gigabitethernet 1/0/1 switchport	Verifies your entries.
Step 8	show running-config Example: Device# show running-config	Verifies your entries.
Step 9	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Monitoring Port Blocking

Table 154: Commands for Displaying Port Blocking Settings

Command	Purpose
show interfaces [interface-id] switchport	Displays the administrative and operational status of all switching (nonrouting) ports or the specified port, including port blocking and port protection settings.

Where to Go Next

Additional References

Related Documents

Related Topic	Document Title

Error Message Decoder

Description	Link
To help you research and resolve system error messages in this release, use the Error Message Decoder tool.	https://www.cisco.com/cgi-bin/Support/Errordecoder/index.cgi

Standards and RFCs

Standard/RFC	Title

MIBs

MB	MIBs Link
	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
<p>The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies.</p> <p>To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds.</p> <p>Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.</p>	http://www.cisco.com/support

Feature Information

Release	Feature Information
Cisco IOS Release 15.0(2)EX1	This feature was introduced.

Prerequisites for Port Security



Note If you try to set the maximum value to a number less than the number of secure addresses already configured on an interface, the command is rejected.

Restrictions for Port Security

The maximum number of secure MAC addresses that you can configure on a switch or switch stack is set by the maximum number of available MAC addresses allowed in the system. This number is determined by the active Switch Database Management (SDM) template. This number is the total of available MAC addresses, including those used for other Layer 2 functions and any other secure MAC addresses configured on interfaces.

Information About Port Security

Port Security

You can use the port security feature to restrict input to an interface by limiting and identifying MAC addresses of the stations allowed to access the port. When you assign secure MAC addresses to a secure port, the port does not forward packets with source addresses outside the group of defined addresses. If you limit the number of secure MAC addresses to one and assign a single secure MAC address, the workstation attached to that port is assured the full bandwidth of the port.

If a port is configured as a secure port and the maximum number of secure MAC addresses is reached, when the MAC address of a station attempting to access the port is different from any of the identified secure MAC addresses, a security violation occurs. Also, if a station with a secure MAC address configured or learned on one secure port attempts to access another secure port, a violation is flagged.

Types of Secure MAC Addresses

The switch supports these types of secure MAC addresses:

- Static secure MAC addresses—These are manually configured by using the **switchport port-security mac-address *mac-address*** interface configuration command, stored in the address table, and added to the switch running configuration.

- Dynamic secure MAC addresses—These are dynamically configured, stored only in the address table, and removed when the switch restarts.
- Sticky secure MAC addresses—These can be dynamically learned or manually configured, stored in the address table, and added to the running configuration. If these addresses are saved in the configuration file, when the switch restarts, the interface does not need to dynamically reconfigure them.

Sticky Secure MAC Addresses

You can configure an interface to convert the dynamic MAC addresses to sticky secure MAC addresses and to add them to the running configuration by enabling sticky learning. The interface converts all the dynamic secure MAC addresses, including those that were dynamically learned before sticky learning was enabled, to sticky secure MAC addresses. All sticky secure MAC addresses are added to the running configuration.

The sticky secure MAC addresses do not automatically become part of the configuration file, which is the startup configuration used each time the switch restarts. If you save the sticky secure MAC addresses in the configuration file, when the switch restarts, the interface does not need to relearn these addresses. If you do not save the sticky secure addresses, they are lost.

If sticky learning is disabled, the sticky secure MAC addresses are converted to dynamic secure addresses and are removed from the running configuration.

Security Violations

It is a security violation when one of these situations occurs:

- The maximum number of secure MAC addresses have been added to the address table, and a station whose MAC address is not in the address table attempts to access the interface.
- An address learned or configured on one secure interface is seen on another secure interface in the same VLAN.
- Running diagnostic tests with port security enabled.

You can configure the interface for one of three violation modes, based on the action to be taken if a violation occurs:

- protect—when the number of secure MAC addresses reaches the maximum limit allowed on the port, packets with unknown source addresses are dropped until you remove a sufficient number of secure MAC addresses to drop below the maximum value or increase the number of maximum allowable addresses. You are not notified that a security violation has occurred.



Note We do not recommend configuring the protect violation mode on a trunk port. The protect mode disables learning when any VLAN reaches its maximum limit, even if the port has not reached its maximum limit.

- restrict—when the number of secure MAC addresses reaches the maximum limit allowed on the port, packets with unknown source addresses are dropped until you remove a sufficient number of secure MAC addresses to drop below the maximum value or increase the number of maximum allowable addresses. In this mode, you are notified that a security violation has occurred. An SNMP trap is sent, a syslog message is logged, and the violation counter increments.

- **shutdown**—a port security violation causes the interface to become error-disabled and to shut down immediately, and the port LED turns off. When a secure port is in the error-disabled state, you can bring it out of this state by entering the **errdisable recovery cause psecure-violation** global configuration command, or you can manually re-enable it by entering the **shutdown** and **no shut down** interface configuration commands. This is the default mode.
- **shutdown vlan**—Use to set the security violation mode per-VLAN. In this mode, the VLAN is error disabled instead of the entire port when a violation occurs

This table shows the violation mode and the actions taken when you configure an interface for port security.

Table 155: Security Violation Mode Actions

Violation Mode	Traffic is forwarded 18	Sends SNMP trap	Sends syslog message	Displays error message 19	Violation counter increments	Shuts down port
protect	No	No	No	No	No	No
restrict	No	Yes	Yes	No	Yes	No
shutdown	No	No	No	No	Yes	Yes
shutdown vlan	No	No	Yes	No	Yes	No 20

¹⁸ Packets with unknown source addresses are dropped until you remove a sufficient number of secure MAC addresses.

¹⁹ The switch returns an error message if you manually configure an address that would cause a security violation.

²⁰ Shuts down only the VLAN on which the violation occurred.

Port Security Aging

You can use port security aging to set the aging time for all secure addresses on a port. Two types of aging are supported per port:

- **Absolute**—The secure addresses on the port are deleted after the specified aging time.
- **Inactivity**—The secure addresses on the port are deleted only if the secure addresses are inactive for the specified aging time.

Port Security and Switch Stacks

When a switch joins a stack, the new switch will get the configured secure addresses. All dynamic secure addresses are downloaded by the new stack member from the other stack members.

When a switch (either the active switch or a stack member) leaves the stack, the remaining stack members are notified, and the secure MAC addresses configured or learned by that switch are deleted from the secure MAC address table.

Default Port Security Configuration

Table 156: Default Port Security Configuration

Feature	Default Setting
Port security	Disabled on a port.
Sticky address learning	Disabled.
Maximum number of secure MAC addresses per port	1.
Violation mode	Shutdown. The port shuts down when the maximum number of secure MAC addresses is exceeded.
Port security aging	Disabled. Aging time is 0. Static aging is disabled. Type is absolute.

Port Security Configuration Guidelines

- Port security can only be configured on static access ports or trunk ports. A secure port cannot be a dynamic access port.
- A secure port cannot be a destination port for Switched Port Analyzer (SPAN).



Note Voice VLAN is only supported on access ports and not on trunk ports, even though the configuration is allowed.

- When you enable port security on an interface that is also configured with a voice VLAN, set the maximum allowed secure addresses on the port to two. When the port is connected to a Cisco IP phone, the IP phone requires one MAC address. The Cisco IP phone address is learned on the voice VLAN, but is not learned on the access VLAN. If you connect a single PC to the Cisco IP phone, no additional MAC addresses are required. If you connect more than one PC to the Cisco IP phone, you must configure enough secure addresses to allow one for each PC and one for the phone.
- When a trunk port configured with port security and assigned to an access VLAN for data traffic and to a voice VLAN for voice traffic, entering the **switchport voice** and **switchport priority extend** interface configuration commands has no effect.

When a connected device uses the same MAC address to request an IP address for the access VLAN and then an IP address for the voice VLAN, only the access VLAN is assigned an IP address.
- When you enter a maximum secure address value for an interface, and the new value is greater than the previous value, the new value overwrites the previously configured value. If the new value is less than the previous value and the number of configured secure addresses on the interface exceeds the new value, the command is rejected.
- The switch does not support port security aging of sticky secure MAC addresses.

This table summarizes port security compatibility with other port-based features.

Table 157: Port Security Compatibility with Other Switch Features

Type of Port or Feature on Port	Compatible with Port Security
DTP ²¹ port ²²	No
Trunk port	Yes
Dynamic-access port ²³	No
Routed port	No
SPAN source port	Yes
SPAN destination port	No
EtherChannel	Yes
Tunneling port	Yes
Protected port	Yes
IEEE 802.1x port	Yes
Voice VLAN port ²⁴	Yes
IP source guard	Yes
Dynamic Address Resolution Protocol (ARP) inspection	Yes
Flex Links	Yes

²¹ DTP=Dynamic Trunking Protocol

²² A port configured with the **switchport mode dynamic** interface configuration command.

²³ A VLAN Query Protocol (VQP) port configured with the **switchport access vlan dynamic** interface configuration command.

²⁴ You must set the maximum allowed secure addresses on the port to two plus the maximum number of secure addresses allowed on the access VLAN.

Overview of Port-Based Traffic Control

Port-based traffic control is a set of Layer 2 features on the Cisco Catalyst switches used to filter or block packets at the port level in response to specific traffic conditions. The following port-based traffic control features are supported in the Cisco IOS Release for which this guide is written:

- Storm Control
- Protected Ports
- Port Blocking
- Port Security

- Protocol Storm Protection

How to Configure Port Security

Enabling and Configuring Port Security

Before you begin

This task restricts input to an interface by limiting and identifying MAC addresses of the stations allowed to access the port:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **port-security mac-address forbidden** *mac address*
4. **interface** *interface-id*
5. **switchport mode** {access | trunk}
6. **switchport voice vlan** *vlan-id*
7. **switchport port-security**
8. **switchport port-security** [maximum *value* **vlan** {*vlan-list* | {access | voice}}]]
9. **switchport port-security violation** {protect | restrict | shutdown | shutdown vlan}
10. **switchport port-security** [mac-address *mac-address* [**vlan** {*vlan-id* | {access | voice}}]]
11. **switchport port-security mac-address sticky**
12. **switchport port-security mac-address sticky** [*mac-address* | **vlan** {*vlan-id* | {access | voice}}]]
13. **switchport port-security mac-address forbidden** *mac address*
14. **end**
15. **show port-security**
16. **show running-config**
17. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.

	Command or Action	Purpose
Step 3	<p>port-security mac-address forbidden <i>mac address</i></p> <p>Example:</p> <pre>Device(config)# port-security mac-address forbidden 2.2.2</pre>	Specifies a MAC address that should be forbidden by port-security on all the interfaces.
Step 4	<p>interface <i>interface-id</i></p> <p>Example:</p> <pre>Device(config)# interface gigabitethernet1/0/1</pre>	Specifies the interface to be configured, and enter interface configuration mode.
Step 5	<p>switchport mode {access trunk}</p> <p>Example:</p> <pre>Device(config-if)# switchport mode access</pre>	Sets the interface switchport mode as access or trunk; an interface in the default mode (dynamic auto) cannot be configured as a secure port.
Step 6	<p>switchport voice vlan <i>vlan-id</i></p> <p>Example:</p> <pre>Device(config-if)# switchport voice vlan 22</pre>	Enables voice VLAN on a port. vlan-id—Specifies the VLAN to be used for voice traffic.
Step 7	<p>switchport port-security</p> <p>Example:</p> <pre>Device(config-if)# switchport port-security</pre>	Enable port security on the interface. Note Under certain conditions, when port security is enabled on the member ports in a switch stack, the DHCP and ARP packets would be dropped. To resolve this, configure a shut and no shut on the interface.
Step 8	<p>switchport port-security [maximum value [vlan {<i>vlan-list</i> {access voice}}]]</p> <p>Example:</p> <pre>Device(config-if)# switchport port-security maximum 20</pre>	<p>(Optional) Sets the maximum number of secure MAC addresses for the interface. The maximum number of secure MAC addresses that you can configure on a switch or switch stack is set by the maximum number of available MAC addresses allowed in the system. This number is set by the active Switch Database Management (SDM) template. This number is the total of available MAC addresses, including those used for other Layer 2 functions and any other secure MAC addresses configured on interfaces.</p> <p>(Optional) vlan—sets a per-VLAN maximum value</p> <p>Enter one of these options after you enter the vlan keyword:</p> <ul style="list-style-type: none"> • <i>vlan-list</i>—On a trunk port, you can set a per-VLAN maximum value on a range of VLANs separated by

	Command or Action	Purpose
		<p>a hyphen or a series of VLANs separated by commas. For nonspecified VLANs, the per-VLAN maximum value is used.</p> <ul style="list-style-type: none"> • access—On an access port, specifies the VLAN as an access VLAN. • voice—On an access port, specifies the VLAN as a voice VLAN. <p>Note The voice keyword is available only if a voice VLAN is configured on a port and if that port is not the access VLAN. If an interface is configured for voice VLAN, configure a maximum of two secure MAC addresses.</p>
<p>Step 9</p>	<p>switchport port-security violation {protect restrict shutdown shutdown vlan}</p> <p>Example:</p> <pre>Device(config-if)# switchport port-security violation restrict</pre>	<p>(Optional) Sets the violation mode, the action to be taken when a security violation is detected, as one of these:</p> <ul style="list-style-type: none"> • protect—When the number of port secure MAC addresses reaches the maximum limit allowed on the port, packets with unknown source addresses are dropped until you remove a sufficient number of secure MAC addresses to drop below the maximum value or increase the number of maximum allowable addresses. You are not notified that a security violation has occurred. <p>Note We do not recommend configuring the protect mode on a trunk port. The protect mode disables learning when any VLAN reaches its maximum limit, even if the port has not reached its maximum limit.</p> <ul style="list-style-type: none"> • restrict—When the number of secure MAC addresses reaches the limit allowed on the port, packets with unknown source addresses are dropped until you remove a sufficient number of secure MAC addresses or increase the number of maximum allowable addresses. An SNMP trap is sent, a syslog message is logged, and the violation counter increments. • shutdown—The interface is error-disabled when a violation occurs, and the port LED turns off. An SNMP trap is sent, a syslog message is logged, and the violation counter increments. • shutdown vlan—Use to set the security violation mode per VLAN. In this mode, the VLAN is error disabled instead of the entire port when a violation occurs.

	Command or Action	Purpose
		<p>Note When a secure port is in the error-disabled state, you can bring it out of this state by entering the errdisable recovery cause psecure-violation global configuration command. You can manually re-enable it by entering the shutdown and no shutdown interface configuration commands or by using the clear errdisable interface vlan privileged EXEC command.</p>
<p>Step 10</p>	<p>switchport port-security [mac-address <i>mac-address</i> [vlan {<i>vlan-id</i> {access voice}}]]</p> <p>Example:</p> <pre>Device(config-if)# switchport port-security mac-address 00:A0:C7:12:C9:25 vlan 3 voice</pre>	<p>(Optional) Enters a secure MAC address for the interface. You can use this command to enter the maximum number of secure MAC addresses. If you configure fewer secure MAC addresses than the maximum, the remaining MAC addresses are dynamically learned.</p> <p>Note If you enable sticky learning after you enter this command, the secure addresses that were dynamically learned are converted to sticky secure MAC addresses and are added to the running configuration.</p> <p>(Optional) vlan—sets a per-VLAN maximum value.</p> <p>Enter one of these options after you enter the vlan keyword:</p> <ul style="list-style-type: none"> • vlan-id—On a trunk port, you can specify the VLAN ID and the MAC address. If you do not specify a VLAN ID, the native VLAN is used. • access—On an access port, specifies the VLAN as an access VLAN. • voice—On an access port, specifies the VLAN as a voice VLAN. <p>Note The voice keyword is available only if a voice VLAN is configured on a port and if that port is not the access VLAN. If an interface is configured for voice VLAN, configure a maximum of two secure MAC addresses.</p>
<p>Step 11</p>	<p>switchport port-security mac-address sticky</p> <p>Example:</p> <pre>Device(config-if)# switchport port-security mac-address sticky</pre>	<p>(Optional) Enables sticky learning on the interface.</p>

	Command or Action	Purpose
Step 12	<p>switchport port-security mac-address sticky [<i>mac-address</i> vlan {<i>vlan-id</i> {access voice}}]</p> <p>Example:</p> <pre>Device(config-if)# switchport port-security mac-address sticky 00:A0:C7:12:C9:25 vlan voice</pre>	<p>(Optional) Enters a sticky secure MAC address, repeating the command as many times as necessary. If you configure fewer secure MAC addresses than the maximum, the remaining MAC addresses are dynamically learned, are converted to sticky secure MAC addresses, and are added to the running configuration.</p> <p>Note If you do not enable sticky learning before this command is entered, an error message appears, and you cannot enter a sticky secure MAC address.</p> <p>(Optional) vlan—sets a per-VLAN maximum value.</p> <p>Enter one of these options after you enter the vlan keyword:</p> <ul style="list-style-type: none"> • vlan-id—On a trunk port, you can specify the VLAN ID and the MAC address. If you do not specify a VLAN ID, the native VLAN is used. • access—On an access port, specifies the VLAN as an access VLAN. • voice—On an access port, specifies the VLAN as a voice VLAN. <p>Note The voice keyword is available only if a voice VLAN is configured on a port and if that port is not the access VLAN.</p>
Step 13	<p>switchport port-security mac-address forbidden <i>mac address</i></p> <p>Example:</p> <pre>Device(config-if)# switchport port-security mac-address forbidden 2.2.2</pre>	<p>Specifies a MAC address that should be forbidden by port-security on the particular interface.</p>
Step 14	<p>end</p> <p>Example:</p> <pre>Device(config)# end</pre>	<p>Returns to privileged EXEC mode.</p>
Step 15	<p>show port-security</p> <p>Example:</p> <pre>Device# show port-security</pre>	<p>Verifies your entries.</p>

	Command or Action	Purpose
Step 16	show running-config Example: Device# <code>show running-config</code>	Verifies your entries.
Step 17	copy running-config startup-config Example: Device# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Enabling and Configuring Port Security Aging

Use this feature to remove and add devices on a secure port without manually deleting the existing secure MAC addresses and to still limit the number of secure addresses on a port. You can enable or disable the aging of secure addresses on a per-port basis.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **switchport port-security aging** {static | time *time* | type {absolute | inactivity}}
5. **end**
6. **show port-security** [interface *interface-id*] [address]
7. **show running-config**
8. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> <code>enable</code>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode.
Step 3	interface <i>interface-id</i> Example:	Specifies the interface to be configured, and enter interface configuration mode.

	Command or Action	Purpose
	Device(config)# <code>interface gigabitethernet 1/0/1</code>	
Step 4	<p>switchport port-security aging {static time <i>time</i> type {absolute inactivity}}</p> <p>Example:</p> <pre>Device(config-if)# switchport port-security aging time 120</pre>	<p>Enables or disable static aging for the secure port, or set the aging time or type.</p> <p>Note The switch does not support port security aging of sticky secure addresses.</p> <p>Enter static to enable aging for statically configured secure addresses on this port.</p> <p>For <i>time</i>, specifies the aging time for this port. The valid range is from 0 to 1440 minutes.</p> <p>For type, select one of these keywords:</p> <ul style="list-style-type: none"> • absolute—Sets the aging type as absolute aging. All the secure addresses on this port age out exactly after the time (minutes) specified lapses and are removed from the secure address list. • inactivity—Sets the aging type as inactivity aging. The secure addresses on this port age out only if there is no data traffic from the secure source addresses for the specified time period.
Step 5	<p>end</p> <p>Example:</p> <pre>Device(config)# end</pre>	Returns to privileged EXEC mode.
Step 6	<p>show port-security [<i>interface interface-id</i>] [<i>address</i>]</p> <p>Example:</p> <pre>Device# show port-security interface gigabitethernet 1/0/1</pre>	Verifies your entries.
Step 7	<p>show running-config</p> <p>Example:</p> <pre>Device# show running-config</pre>	Verifies your entries.
Step 8	<p>copy running-config startup-config</p> <p>Example:</p> <pre>Device# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Configuration Examples for Port Security

This example shows how to enable port security on a port and to set the maximum number of secure addresses to 50. The violation mode is the default, no static secure MAC addresses are configured, and sticky learning is enabled.

```
Device(config)# interface gigabitethernet 1/0/1
Device(config-if)# switchport mode access
Device(config-if)# switchport port-security
Device(config-if)# switchport port-security maximum 50
Device(config-if)# switchport port-security mac-address sticky
```

This example shows how to configure a static secure MAC address on VLAN 3 on a port:

```
Device(config)# interface gigabitethernet 1/0/2
Device(config-if)# switchport mode trunk
Device(config-if)# switchport port-security
Device(config-if)# switchport port-security mac-address 0000.0200.0004 vlan 3
```

This example shows how to enable sticky port security on a port, to manually configure MAC addresses for data VLAN and voice VLAN, and to set the total maximum number of secure addresses to 20 (10 for data VLAN and 10 for voice VLAN).

```
Device(config)# interface tengigabitethernet 1/0/1
Device(config-if)# switchport access vlan 21
Device(config-if)# switchport mode access
Device(config-if)# switchport voice vlan 22
Device(config-if)# switchport port-security
Device(config-if)# switchport port-security maximum 20
Device(config-if)# switchport port-security violation restrict
Device(config-if)# switchport port-security mac-address sticky
Device(config-if)# switchport port-security mac-address sticky 0000.0000.0002
Device(config-if)# switchport port-security mac-address 0000.0000.0003
Device(config-if)# switchport port-security mac-address sticky 0000.0000.0001 vlan voice
Device(config-if)# switchport port-security mac-address 0000.0000.0004 vlan voice
Device(config-if)# switchport port-security maximum 10 vlan access
Device(config-if)# switchport port-security maximum 10 vlan voice
```

Additional References

MIBs

MIB	MIBs Link
All the supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
<p>The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies.</p> <p>To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds.</p> <p>Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.</p>	<p>http://www.cisco.com/support</p>

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table at the end of this module.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to <http://www.cisco.com/go/cfn>. An account on Cisco.com is not required.

Information About Protocol Storm Protection

Protocol Storm Protection

When a switch is flooded with Address Resolution Protocol (ARP) or control packets, high CPU utilization can cause the CPU to overload. These issues can occur:

- Routing protocol can flap because the protocol control packets are not received, and neighboring adjacencies are dropped.
- Spanning Tree Protocol (STP) reconverges because the STP bridge protocol data unit (BPDU) cannot be sent or received.
- CLI is slow or unresponsive.

Using protocol storm protection, you can control the rate at which control packets are sent to the switch by specifying the upper threshold for the packet flow rate. The supported protocols are ARP, ARP snooping, Dynamic Host Configuration Protocol (DHCP) v4, DHCP snooping, Internet Group Management Protocol (IGMP), and IGMP snooping.

When the packet rate exceeds the defined threshold, the switch drops all traffic arriving on the specified virtual port for 30 seconds. The packet rate is measured again, and protocol storm protection is again applied if necessary.

For further protection, you can manually error disable the virtual port, blocking all incoming traffic on the virtual port. You can manually enable the virtual port or set a time interval for automatic re-enabling of the virtual port.



Note Excess packets are dropped on no more than two virtual ports.
Virtual port error disabling is not supported for EtherChannel and Flexlink interfaces

Default Protocol Storm Protection Configuration

Protocol storm protection is disabled by default. When it is enabled, auto-recovery of the virtual port is disabled by default.

How to Configure Protocol Storm Protection

Enabling Protocol Storm Protection

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **psp {arp | dhcp | igmp} pps *value***
4. **errdisable detect cause psp**
5. **errdisable recovery interval *time***
6. **end**
7. **show psp config {arp | dhcp | igmp}**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	psp {arp dhcp igmp} pps <i>value</i> Example:	Configures protocol storm protection for ARP, IGMP, or DHCP.

	Command or Action	Purpose
	Device(config)# psp dhcp pps 35	For <i>value</i> , specifies the threshold value for the number of packets per second. If the traffic exceeds this value, protocol storm protection is enforced. The range is from 5 to 50 packets per second.
Step 4	errdisable detect cause psp Example: Device(config)# errdisable detect cause psp	(Optional) Enables error-disable detection for protocol storm protection. If this feature is enabled, the virtual port is error disabled. If this feature is disabled, the port drops excess packets without error disabling the port.
Step 5	errdisable recovery interval time Example: Device	(Optional) Configures an auto-recovery time (in seconds) for error-disabled virtual ports. When a virtual port is error-disabled, the switch auto-recovers after this time. The range is from 30 to 86400 seconds.
Step 6	end Example: Device(config)# end	Returns to privileged EXEC mode.
Step 7	show psp config {arp dhcp igmp} Example: Device# show psp config dhcp	Verifies your entries.

Monitoring Protocol Storm Protection

Command	Purpose
show psp config {arp dhcp igmp}	Verify your entries.

Additional References

MIBs

MIB	MIBs Link
All the supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
<p>The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies.</p> <p>To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds.</p> <p>Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.</p>	http://www.cisco.com/support



CHAPTER 78

Configuring Cisco TrustSec

- [Information about Cisco TrustSec, on page 1617](#)
- [Cisco TrustSec Features, on page 1618](#)
- [Feature Information for Cisco TrustSec, on page 1618](#)

Information about Cisco TrustSec

Cisco TrustSec provides security improvements to Cisco network devices based on the capability to strongly identify users, hosts, and network devices within a network. TrustSec provides topology-independent and scalable access controls by uniquely classifying data traffic for a particular role. TrustSec ensures data confidentiality and integrity by establishing trust among authenticated peers and encrypting links with those peers.

The key component of Cisco TrustSec is the Cisco Identity Services Engine (ISE). Cisco ISE can provision switches with TrustSec Identities and Security Group ACLs (SGACLs), though these may be configured manually on the switch.

MTU Guidelines

CTS tagged packets greater than 1518 bytes may get dropped on the Cisco vWLC controller. This is due to a restriction on the size of incoming packets on the UCS server, which is hosting vWLC instances. The UCS server have a default MTU of 1500 thereby allowing packets of 1518 bytes only. Here, the additional 18 bytes includes 4 bytes of 802.1Q and 14 bytes of Ethernet header.

An Ethernet link configured for CTS tagging imposes a 8-byte encapsulation called Cisco metadata. As a result, the total size of the Ethernet packet is increased by 8 bytes to 1526 bytes ($1518+8 = 1526$). Hence, the MTU of the receiving interface has to be increased by 8-bytes to accommodate the additional 8 bytes in the Ethernet.

While CTS interfaces on the routers and switches (for example, Cisco ASR 1000 Series Routers, Cisco 4000 Series Integrated Services Routers, Cisco Catalyst 3000 Series Switches, Cisco Catalyst 9000 Series Switches) auto-adjusts MTU to 1508 bytes to accommodate additional 8-byte. However, other devices like UCS servers requires manual update to increase the MTU to 1508. For information on how to configure jumbo MTU on UCS, see the following link:

<https://www.cisco.com/c/en/us/support/docs/servers-unified-computing/ucs-b-series-blade-servers/117601-configure-UCS-00.html>

Cisco TrustSec Features

The table below lists the Cisco TrustSec features implemented on Cisco TrustSec-enabled Catalyst 2960-X and 2960-XR Series Switches:

Cisco TrustSec Feature	Description
Endpoint Admission Control (EAC)	EAC is an authentication process for an endpoint user or a device connecting to the TrustSec domain. Usually EAC takes place at the access level switch. Successful authentication and authorization in the EAC process results in Security Group Tag assignment for the user or device. Currently EAC can be 802.1X, MAC Authentication Bypass (MAB), and Web Authentication Proxy (WebAuth).
SGT Exchange Protocol (SXP)	Security Group Tag Exchange Protocol (SXP). With SXP, devices that are not TrustSec-hardware-capable can receive SGT attributes for authenticated users and devices from the Cisco Identity Services Engine (ISE) or the Cisco Secure Access Control System (ACS). The devices can then forward a sourceIP-to-SGT binding to a TrustSec-hardware-capable device will tag the source traffic for SGACL enforcement.

Feature Information for Cisco TrustSec

Table 158: Feature Information for Cisco TrustSec

Feature Name	Release	Feature Information
SXPv1 and SXPv2	Cisco IOS XE 15.0(2)EX1	SXP is introduced on the Catalyst 2960-XR switch.



CHAPTER 79

Configuring FIPS

- [Information About FIPS and Common Criteria, on page 1619](#)

Information About FIPS and Common Criteria

The Federal Information Processing Standard (FIPS) certification documents for Cisco Catalyst series switches are posted on the following website:

http://www.cisco.com/web/strategy/government/security_certification/net_business_benefit_seccert_fips140.html

Click the link in the Certification column to view the Consolidated Validation Certificate and the Security Policy document. The Security Policy document describes the FIPS implementation, hardware installation, firmware initialization, and software configuration procedures for FIPS operation.

Common Criteria is an international standard (ISO/IEC 15408) for computer security certification. This standard is a set of requirements, tests, and evaluation methods that ensures that the Target of Evaluation complies with a specific Protection Profile or custom Security Target. For more information, see the security target document for specific Cisco Catalyst switch models and Cisco IOS Releases at: http://www.niap-ccevs.org/CCEVS_Products/pcl.cfm?tech_name=Network+Switch



CHAPTER 80

Configuring Control Plane Policing

- [Restrictions for Control Plane Policing, on page 1621](#)
- [Control Plane Policing, on page 1621](#)
- [Configuring Control Plane Policing, on page 1622](#)
- [Examples: Configuring CoPP, on page 1623](#)

Restrictions for Control Plane Policing

The following restrictions apply while Configuring Control Plane Policing:

- Only six among the following protocols can be configured simultaneously: **rip**, **ospf-v6**, **eigrp-v6**, **rip-v6**, **dhcp-snoop-client-to-server**, **dhcp-snoop-server-to-client**, **ndp-router-solicitation**, **ndp-router-advertisement**, **ndp-redirect**, **dhcpv6-client-to-server**, **dhcpv6-server-to-client**, **igrp**.
- For **ospf**, **eigrp** and **ripv2** protocols, control packets which are destined to multicast Mac of the router are policed along with the "**reserve-multicast-group**" option.

Control Plane Policing

Configure the Control Plane Policing (CoPP) feature on a predefined set of protocols to control the flow of traffic coming to the CPU. The CoPP allows you to set a rate limit on specific protocol packets. These packets are policed, and the packets that conform to the defined rate limit are permitted into the CPU. CoPP protects the packets from being routed to the CPU at an undesired rate that might impact the performance of a switch and the network. In addition, the CoPP protects the CPU from denial of service (DoS) attacks and ensures routing stability, reachability, and packet delivery. You can use Multi-Layer Switching QoS CLI to set the rate limit and policing parameters on a specific protocol.



Note CoPP is supported only on LAN BASE, IP Lite, and IP Service licenses.

Configuring Control Plane Policing

Configure the Control Plane Policing (CoPP) feature on a predefined set of protocols to control the flow of traffic coming into the CPU.

SUMMARY STEPS

1. enable
2. configure terminal
3. mls qos copp protocol { autorp-announce | autorp-discovery | bgp | cdp | cgmp | dai | dhcp-snoop-client-to-server | dhcp-snoop-server-to-client | dhcpv6-client-to-server | dhcpv6-server-to-client | eigrp | eigrp-v6 | energy-wise | igmp-gs-query | igmp-leave | igmp-query | igmp-report | igmp | ipv6-pimv2 | lldp | mld-gs-query | mld-leave | mld-query | mld-report | ndp-redirect | ndp-router-advertisement | ndp-router-solicitation | ospf | ospf-v6 | pimv1 | pxe | rep-hfl | reserve-multicast-group | rip | rip-v6 | rsvp-snoop | stp } police {pps | bps} police rate
4. end
5. show mls qos copp protocols
6. copy running-config startup-config

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	mls qos copp protocol { autorp-announce autorp-discovery bgp cdp cgmp dai dhcp-snoop-client-to-server dhcp-snoop-server-to-client dhcpv6-client-to-server dhcpv6-server-to-client eigrp eigrp-v6 energy-wise igmp-gs-query igmp-leave igmp-query igmp-report igmp ipv6-pimv2 lldp mld-gs-query mld-leave mld-query mld-report ndp-redirect ndp-router-advertisement ndp-router-solicitation ospf ospf-v6 pimv1 pxe rep-hfl reserve-multicast-group rip rip-v6 rsvp-snoop stp } police {pps bps} police rate Example: Device (config)# mls qos copp protocol cdp police	Configures a packet policer for the specified protocol. For more details about the various parameters, please refer <i>Consolidated Platform Command Reference, Cisco IOS Release 15.2(4)E</i> .

	Command or Action	Purpose
	<pre>bps 10000 Device(config)# mls qos copp protocol cdp police pps 500</pre>	
Step 4	<p>end</p> <p>Example:</p> <pre>Device(config)# end</pre>	Returns to privileged EXEC mode.
Step 5	<p>show mls qos copp protocols</p> <p>Example:</p> <pre>Device# show mls qos copp protocols</pre>	Displays the CoPP parameters and counters for all the configured protocol.
Step 6	<p>copy running-config startup-config</p> <p>Example:</p> <pre>Device# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

What to do next

To clear the CoPP statistics, use the **clear copp counters** command.

Examples: Configuring CoPP

The following example shows how to enable Control Plane Policing (CoPP) for a specific protocol:

```
Switch (config)# mls qos copp protocol cdp police bps ?
<8000-2000000000> Bits per second (postfix k, m, g optional; decimal point allowed)
Switch (config)# mls qos copp protocol cdp police bps 10000
Switch(config)# mls qos copp protocol cdp police pps ?
<100-100000> Packet per second
Switch(config)# mls qos copp protocol cdp police pps 500
```

The following example shows the CoPP parameters and counters for all the configured protocol:

```
Switch# show running-config | inc copp
Switch#show running-config | inc copp
mls qos copp protocol rep-hfl police pps 5600
mls qos copp protocol lldp police bps 908900
mls qos copp protocol cdp police pps 3434
```

/* Copp detailed output */

```
Switch#show mls qos copp protocols
```

```
-----
Protocol                Mode      PolicerRate      PolicerBurst
InProfilePackets      OutProfilePackets  InProfileBytes    OutProfileBytes
-----
rep-hfl                 pps      5600              5600
0                       0         0                 0
-----
```

lldp		bps	908900	908900
0	0		0	0
cdp		pps	3434	3434
45172	0		2891008	0



PART **XII**

Configuring Cisco IOS IP SLAs

- [Configuring Cisco IP SLAs, on page 1627](#)



CHAPTER 81

Configuring Cisco IP SLAs

- [Restrictions on SLAs, on page 1627](#)
- [Information About SLAs, on page 1627](#)
- [How to Configure IP SLAs Operations, on page 1631](#)
- [Monitoring IP SLA Operations, on page 1632](#)
- [Additional References, on page 1633](#)
- [Feature History and Information for Service Level Agreements, on page 1634](#)

Restrictions on SLAs

This section lists the restrictions on SLAs.

The following are restrictions on IP SLAs network performance measurement:

- The device does not support VoIP service levels using the gatekeeper registration delay operations measurements.
- Only a Cisco IOS device can be a source for a destination IP SLAs responder.
- You cannot configure the IP SLAs responder on non-Cisco devices and Cisco IOS IP SLAs can send operational packets only to services native to those devices.

Information About SLAs

Cisco IOS IP Service Level Agreements (SLAs)

Cisco IOS IP SLAs send data across the network to measure performance between multiple network locations or across multiple network paths. They simulate network data and IP services and collect network performance information in real time. Cisco IOS IP SLAs generate and analyze traffic either between Cisco IOS devices or from a Cisco IOS device to a remote IP device such as a network application server. Measurements provided by the various Cisco IOS IP SLA operations can be used for troubleshooting, for problem analysis, and for designing network topologies.

Depending on the specific Cisco IOS IP SLA operations, various network performance statistics are monitored within the Cisco device and stored in both command-line interface (CLI) and Simple Network Management Protocol (SNMP) MIBs. IP SLA packets have configurable IP and application layer options such as source

and destination IP address, User Datagram Protocol (UDP)/TCP port numbers, a type of service (ToS) byte (including Differentiated Services Code Point [DSCP] and IP Prefix bits), Virtual Private Network (VPN) routing/forwarding instance (VRF), and URL web address.

Because Cisco IP SLAs are Layer 2 transport independent, you can configure end-to-end operations over disparate networks to best reflect the metrics that an end user is likely to experience. IP SLAs collect and analyze the following performance metrics:

- Delay (both round-trip and one-way)
- Jitter (directional)
- Packet loss (directional)
- Packet sequencing (packet ordering)
- Path (per hop)
- Connectivity (directional)
- Server or website download time

Because Cisco IOS IP SLAs is SNMP-accessible, it can also be used by performance-monitoring applications like Cisco Prime Internetwork Performance Monitor (IPM) and other third-party Cisco partner performance management products.

Using IP SLAs can provide the following benefits:

- Service-level agreement monitoring, measurement, and verification.
- Network performance monitoring
 - Measurement of jitter, latency, or packet loss in the network.
 - Continuous, reliable, and predictable measurements.
- IP service network health assessment to verify that the existing QoS is sufficient for new IP services.
- Edge-to-edge network availability monitoring for proactive verification and connectivity testing of network resources (for example, shows the network availability of an NFS server used to store business critical data from a remote site).
- Network operation troubleshooting by providing consistent, reliable measurement that immediately identifies problems and saves troubleshooting time.
- Multiprotocol Label Switching (MPLS) performance monitoring and network verification (if the device supports MPLS).

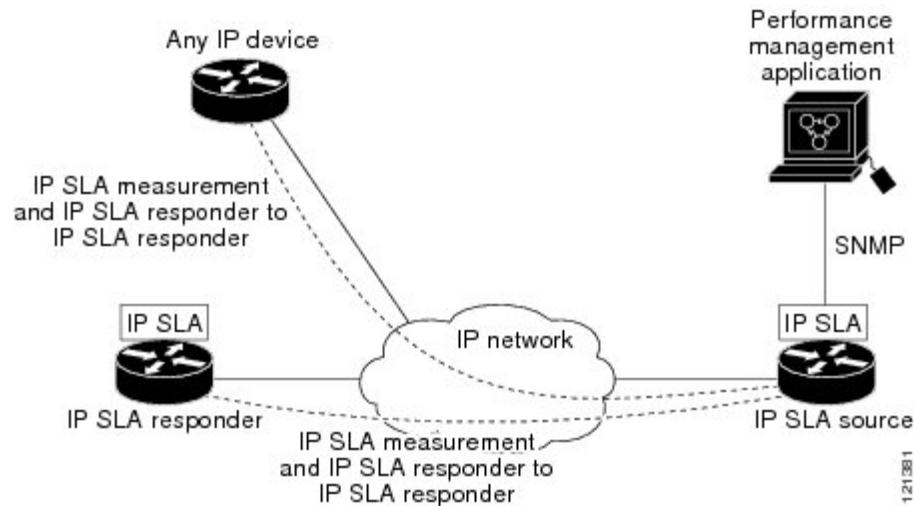
Network Performance Measurement with Cisco IOS IP SLAs

You can use IP SLAs to monitor the performance between any area in the network—core, distribution, and edge—without deploying a physical probe. It uses generated traffic to measure network performance between two networking devices.

Figure 123: Cisco IOS IP SLAs Operation

The following figure shows how IP SLAs begin when the source device sends a generated packet to the destination device. After the destination device receives the packet, depending on the type of IP SLAs operation,

it responds with time-stamp information for the source to make the calculation on performance metrics. An IP SLAs operation performs a network measurement from the source device to a destination in the network using a specific protocol such as UDP.



IP SLA Responder and IP SLA Control Protocol

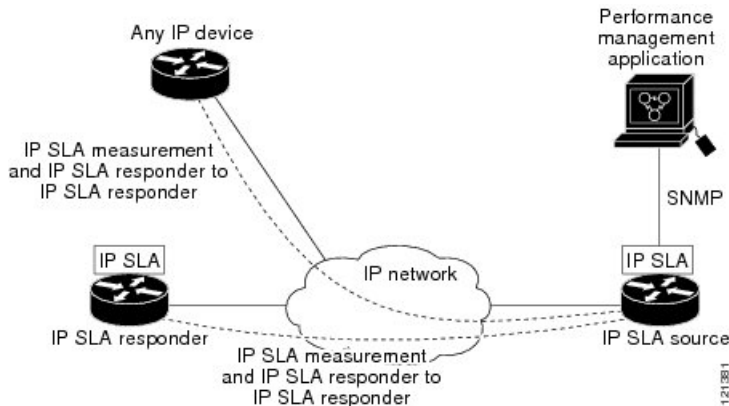
The IP SLA responder is a component embedded in the destination Cisco device that allows the system to anticipate and respond to IP SLA request packets. The responder provides accurate measurements without the need for dedicated probes. The responder uses the Cisco IOS IP SLA Control Protocol to provide a mechanism through which it can be notified on which port it should listen and respond.



Note The IP SLA responder can be a Cisco IOS Layer 2, responder-configurable device. The responder does not need to support full IP SLA functionality.

The following figure shows where the Cisco IOS IP SLA responder fits in the IP network. The responder listens on a specific port for control protocol messages sent by an IP SLA operation. Upon receipt of the control message, it enables the specified UDP or TCP port for the specified duration. During this time, the responder accepts the requests and responds to them. It disables the port after it responds to the IP SLA packet, or when the specified time expires. MD5 authentication for control messages is available for added security.

Figure 124: Cisco IOS IP SLAs Operation



You do not need to enable the responder on the destination device for all IP SLA operations. For example, a responder is not required for services that are already provided by the destination router (such as Telnet or HTTP).

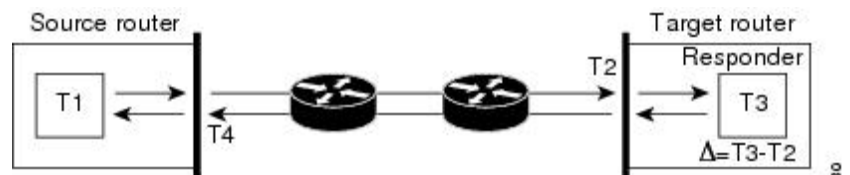
Response Time Computation for IP SLAs

Switches, controllers, and routers can take tens of milliseconds to process incoming packets due to other high priority processes. This delay affects the response times because the test-packet reply might be in a queue while waiting to be processed. In this situation, the response times would not accurately represent true network delays. IP SLAs minimize these processing delays on the source device as well as on the target device (if the responder is being used) to determine true round-trip times. IP SLA test packets use time stamping to minimize the processing delays.

When the IP SLA responder is enabled, it allows the target device to take time stamps when the packet arrives on the interface at interrupt level and again just as it is leaving, eliminating the processing time. This time stamping is made with a granularity of sub-milliseconds (ms).

Figure 125: Cisco IOS IP SLA Responder Time Stamping

The following figure demonstrates how the responder works. Four time stamps are taken to make the calculation for round-trip time. At the target router, with the responder functionality enabled, time stamp 2 (TS2) is subtracted from time stamp 3 (TS3) to produce the time spent processing the test packet as represented by delta. This delta value is then subtracted from the overall round-trip time. Notice that the same principle is applied by IP SLAs on the source router where the incoming time stamp 4 (TS4) is also taken at the interrupt



level to allow for greater accuracy. RTT (Round-trip time) = $T4$ (Time stamp 4) - $T1$ (Time stamp 1) - Δ

An additional benefit of the two time stamps at the target device is the ability to track one-way delay, jitter, and directional packet loss. Because much network behavior is asynchronous, it is critical to have these statistics. However, to capture one-way delay measurements, you must configure both the source router and target router with Network Time Protocol (NTP) so that the source and target are synchronized to the same clock source. One-way jitter measurements do not require clock synchronization.

How to Configure IP SLAs Operations

This section does not include configuration information for all available operations as the configuration information details are included in the *Cisco IOS IP SLAs Configuration Guide*. It does include several operations as examples, including configuring the responder, configuring a UDP jitter operation, which requires a responder, and configuring an ICMP echo operation, which does not require a responder. For details about configuring other operations, see the *Cisco IOS IP SLAs Configuration Guide*.

Default Configuration

No IP SLAs operations are configured.

Configuration Guidelines

For information on the IP SLA commands, see the *Cisco IOS IP SLAs Command Reference, Release 12.4T* command reference.

For detailed descriptions and configuration procedures, see the *Cisco IOS IP SLAs Configuration Guide, Release 12.4TL*.

Configuring the IP SLA Responder

The IP SLA responder is available only on Cisco IOS software-based devices, including some Layer 2 devices that do not support full IP SLA functionality.

Follow these steps to configure the IP SLA responder on the target device (the operational target):

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip sla responder {tcp-connect | udp-echo} ipaddress ip-address port port-number**
4. **end**
5. **show running-config**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example:	Enters global configuration mode.

	Command or Action	Purpose
	Device# <code>configure terminal</code>	
Step 3	<p>ip sla responder {tcp-connect udp-echo} ipaddress <i>ip-address</i> port <i>port-number</i></p> <p>Example:</p> <pre>Device(config)# ip sla responder udp-echo 172.29.139.134 5000</pre>	<p>Configures the device as an IP SLA responder.</p> <p>The keywords have these meanings:</p> <ul style="list-style-type: none"> • tcp-connect—Enables the responder for TCP connect operations. • udp-echo—Enables the responder for User Datagram Protocol (UDP) echo or jitter operations. • ipaddress <i>ip-address</i>—Enter the destination IP address. • port <i>port-number</i>—Enter the destination port number. <p>Note The IP address and port number must match those configured on the source device for the IP SLA operation.</p>
Step 4	<p>end</p> <p>Example:</p> <pre>Device(config)# end</pre>	Returns to privileged EXEC mode.
Step 5	<p>show running-config</p> <p>Example:</p> <pre>Device# show running-config</pre>	Verifies your entries.
Step 6	<p>copy running-config startup-config</p> <p>Example:</p> <pre>Device# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Monitoring IP SLA Operations

The following table describes the commands used to display IP SLA operation configurations and results:

Table 159: Monitoring IP SLA Operations

show ip sla authentication	Displays IP SLA authentication information.
show ip sla responder	Displays information about the IP SLA responder.

Additional References

Related Documents

Related Topic	Document Title
Cisco Medianet Metadata Guide	http://www.cisco.com/c/en/us/td/docs/ios-xml/ios/mdata/configuration/15-sy/mdata-15sy-book/metadata-framework.pdf
Cisco Media Services Proxy Configuration Guide	http://www.cisco.com/c/en/us/td/docs/ios-xml/ios/msp/configuration/15-mt/msp-15-mt-book.pdf
Cisco Mediatrace and Cisco Performance Monitor Configuration Guide	http://www.cisco.com/c/en/us/td/docs/ios-xml/ios/media_monitoring/configuration/15-mt/mm-15-mt-book/mm-mediatrace.html

Error Message Decoder

Description	Link
To help you research and resolve system error messages in this release, use the Error Message Decoder tool.	https://www.cisco.com/cgi-bin/Support/Errordecoder/index.cgi

Standards and RFCs

Standard/RFC	Title
None	-

MIBs

MIB	MIBs Link
All supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
<p>The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies.</p> <p>To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds.</p> <p>Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.</p>	http://www.cisco.com/support

Feature History and Information for Service Level Agreements

Release	Modification
Cisco IOS Release 15.0(2)EX1	This feature was introduced.



PART **XIII**

Stacking

- [Managing Switch Stacks, on page 1637](#)
- [FlexStack-Extended, on page 1665](#)



CHAPTER 82

Managing Switch Stacks

- [Prerequisites for Switch Stacks, on page 1637](#)
- [Restrictions for Switch Stacks, on page 1637](#)
- [Information About Switch Stacks, on page 1638](#)
- [How to Configure a Switch Stack, on page 1651](#)
- [Troubleshooting the Switch Stack, on page 1657](#)
- [Monitoring the Device Stack, on page 1659](#)
- [Configuration Examples for Switch Stacks, on page 1660](#)
- [Additional References for Switch Stacks, on page 1662](#)

Prerequisites for Switch Stacks

All stack members must run the same Cisco IOS software image to ensure compatibility among stack members. For switch stack hardware considerations, see the *Catalyst 2960-XR Switch Hardware Installation Guide*.

Restrictions for Switch Stacks

The following are restrictions for your switch stack configuration:

- The switch does not support mixed stacking. Stacking is supported only with other Catalyst 2960-XR switches.
- Auto-upgrade of stack can not be done when one of the switches in stack is with version Cisco IOS 15.2(3)E. This means that whenever any of the switches in the stack goes into a version mismatch, and if either the active stack is running Cisco IOS 15.2(3)E, or if a member is running Cisco 15.2(3)E, the member can not be auto-upgraded to the required version.

Information About Switch Stacks

Switch Stack Overview

A switch stack is a set of up to eight stacking-capable switches connected through their stack ports. You can connect only one switch type in a stack, or you can connect a mix of Catalyst 2960-XR, Catalyst 2960-X and Catalyst 2960-S switches in the stack. The stack can have one of these configurations:

- Homogeneous stack—A Catalyst 2960-XR only stack with only Catalyst 2960-XR switches as stack members. A homogenous stack can have up to 8 stack members.
- Mixed stack—A stack with a mix of Catalyst 2960-XR, Catalyst 2960-X and Catalyst 2960-S switches as stack members. A mixed stack of Catalyst 2960-XR and Catalyst 2960-X switches can have up to 8 stack members. A mixed stack that includes a Catalyst 2960-S switch can have up to 4 stack members.

The active stack controls the operation of the switch stack, and is the single point of stack-wide management. From the active stack, you configure:

- System-level (global) features that apply to all stack members
- Interface-level features for each stack member

The active stack contains the saved and running configuration files for the switch stack. The configuration files include the system-level settings for the switch stack and the interface-level settings for each stack member. Each stack member has a current copy of these files for back-up purposes.

Supported Features in a Switch Stack

The system-level features supported on the active switch are supported on the entire switch stack.

Encryption Features

If the active switch is running the cryptographic universal software image (supports encryption), the encryption features are available on the switch stack.

FlexStack-Plus

The stack members use the Cisco FlexStack-Plus technology to work together as a unified system. Layer 2 protocols support the entire switch stack as a single entity in the network.



Note Switch stacks running the LAN Base image do not support Layer 3 features.

The FlexStack-Plus bandwidth for a single stack port is 20 Gbps. With FlexStack-Plus technology, up to eight members can be joined into a single stack. In a mixed stack of Catalyst 2960-X and Catalyst 2960-S switches, FlexStack-Plus reverts to FlexStack capabilities of 10 Gbps stack port bandwidth and a maximum of four members per stack.

Switch Stack Membership

A switch stack has up to eight stack members connected through their stack ports. A switch stack always has one active switch.

A standalone device is a device stack with one stack member that also operates as the active switch. You can connect one standalone device to another to create a stack containing two stack members, with one of them as the active switch. You can connect standalone devices to an existing device stack to increase the stack membership.

Figure 126: Creating a Switch Stack from Two Standalone Switches

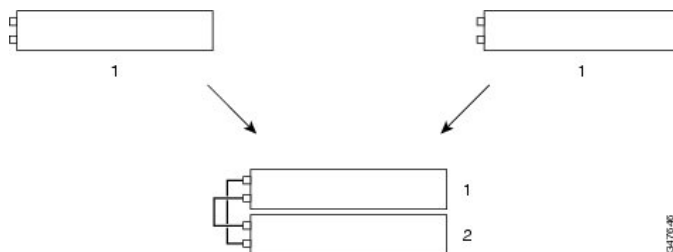
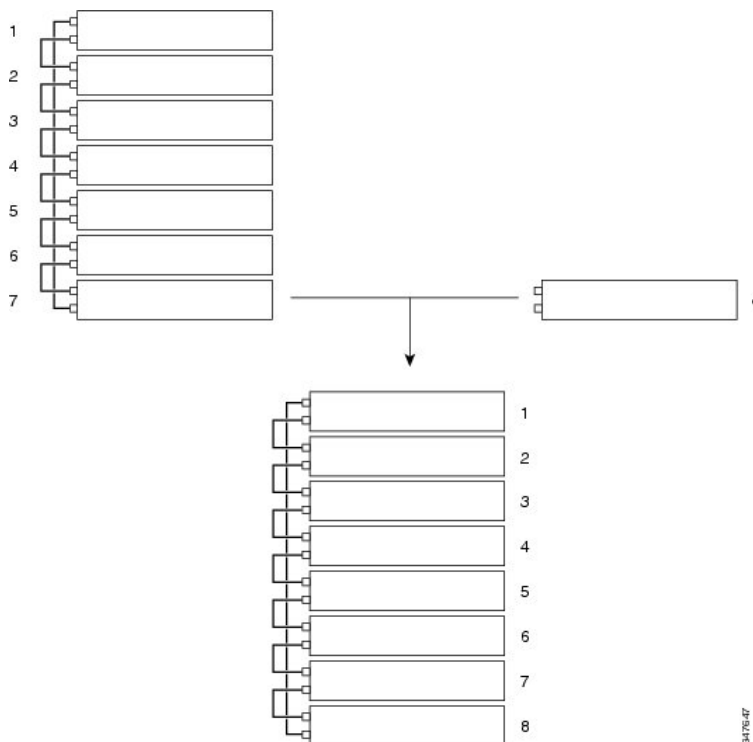


Figure 127: Adding a Standalone Switch to a Switch Stack



Changes to Switch Stack Membership

If you replace a stack member with an identical model, the new switch functions with exactly the same configuration as the replaced switch, assuming that the new switch (referred to as the provisioned switch) is using the same member number as the replaced switch.

The operation of the switch stack continues uninterrupted during membership changes unless you remove the active switch or you add powered-on standalone switches or switch stacks.

- Adding powered-on switches (merging) causes all switches to reload and elect a new active switch from among themselves. The newly elected active switch retains its role and configuration. All other switches retain their stack member numbers and use the stack configuration of the newly elected active switch.



Note In Cisco IOS XE 3.6.4E and later versions, when a new switch is powered-on as a standalone switch before it is added as part of the switch stack, only this switch is reloaded and not the whole switch stack.

Adding powered-on switches (merging) causes the active stack of the merging switch stacks to elect an active stack from among themselves. The reelected active stack retains its role and configuration as do its stack members. All remaining switches, including the former active stacks, reload and join the switch stack as stack members. They change their stack member numbers to the lowest available numbers and use the stack configuration of the reelected active stack.

- Removing powered-on stack members causes the switch stack to divide (partition) into two or more switch stacks, each with the same configuration. This can cause:
 - An IP address conflict in your network. If you want the switch stacks to remain separate, change the IP address or addresses of the newly created switch stacks.
 - A MAC address conflict between two members in the stack. You can use the **stack-mac update force** command to resolve the conflict.

If a newly created switch stack does not have an active switch or standby switch, the switch stack will reload and elect a new active switch.



Note Make sure that you power off the switches that you add to or remove from the switch stack.

After adding or removing stack members, make sure that the switch stack is operating at full bandwidth (40 Gbps). Press the Mode button on a stack member until the Stack mode LED is on. The last two right port LEDs on all switches in the stack should be green. Depending on the switch model, the last two right ports are 10-Gigabit Ethernet ports or small form-factor pluggable (SFP) module ports (10/100/1000 ports). If one or both of these LEDs are not green on any of the switches, the stack is not operating at full bandwidth.

It may take upto 4 seconds for stack convergence when a new stack member is added to the existing switch stack.

If you remove powered-on members but do not want to partition the stack:

- Power off the switches in the newly created switch stacks.
- Reconnect them to the original switch stack through their stack ports.
- Power on the switches.

For cabling and power considerations that affect switch stacks, see the *Catalyst 2960-XR Switch Hardware Installation Guide*.

Stack Member Numbers

The stack member number (1 to 8) identifies each member in the switch stack. The member number also determines the interface-level configuration that a stack member uses. You can display the stack member number by using the **show switch EXEC** command.

A new, out-of-the-box device (one that has not joined a device stack or has not been manually assigned a stack member number) ships with a default stack member number of 1. When it joins a device stack, its default stack member number changes to the lowest available member number in the stack.

Stack members in the same stack cannot have the same stack member number. Every stack member, including a standalone device, retains its member number until you manually change the number or unless the number is already being used by another member in the stack.

- If you manually change the stack member number by using the **switch current-stack-member-number renumber new-stack-member-number** command, the new number goes into effect after that stack member resets (or after you use the **reload slot stack-member-number** privileged EXEC command) and only if that number is not already assigned to any other members in the stack. Another way to change the stack member number is by changing the device_NUMBER environment variable.

If the number is being used by another member in the stack, the device selects the lowest available number in the stack.

If you manually change the number of a stack member and no interface-level configuration is associated with that new member number, that stack member resets to its default configuration.

You cannot use the **switch current-stack-member-number renumber new-stack-member-number** command on a provisioned device. If you do, the command is rejected.

- If you move a stack member to a different device stack, the stack member retains its number only if the number is not being used by another member in the stack. If it is being used, the device selects the lowest available number in the stack.
- If you merge device stacks, the device that join the device stack of a new active device select the lowest available numbers in the stack.

As described in the hardware installation guide, you can use the device port LEDs in Stack mode to visually determine the stack member number of each stack member.

In the **default** mode Stack LED will blink in green color only on the active switch. However, when we scroll the Mode button to **Stack** option - Stack LED will glow green on all the stack members.

When mode button is scrolled to **Stack** option, the switch number of each stack member will be displayed as LEDs on the first five ports of that switch. The switch number is displayed in binary format for all stack members. On the switch, the amber LED indicates value 0 and green LED indicates value 1.

Example for switch number 5 (Binary - 00101):

First five LEDs glow as follows on stack member with switch number 5.

- Port-1 : Amber
- Port-2 : Amber
- Port-3 : Green
- Port-4 : Amber
- Port-5 : Green

Similarly, the first five LEDs glow amber or green, depending on the switch number on all stack members.



- Note**
- Stack port will not go down but only transmission/reception will be disabled. The log message shown below will be displayed on the console. Once the peer end network port is converted to stack port, transmission/reception on this stack port will be enabled.

```
%STACKMGR-4-HSTACK_LINK_CONFIG: Verify peer stack port setting for hstack
StackPort-1 switch 5 (hostname-switchnumber)
```

Stack Member Priority Values

A higher priority value for a stack member increases the probability of it being elected active switch and retaining its stack member number. The priority value can be 1 to 15. The default priority value is 1. You can display the stack member priority value by using the **show switch EXEC** command.



- Note** We recommend assigning the highest priority value to the device that you prefer to be the active device. This ensures that the device is reelected as the active device if a reelection occurs.

To change the priority value for a stack member, use the **switch stack-member-number priority new priority-value** command. For more information, see the “Setting the Stack Member Priority Value” section.

The new priority value takes effect immediately but does not affect the current active device. The new priority value helps determine which stack member is elected as the new active device when the current active device or the device stack resets.

Switch Stack Bridge ID and MAC Address

The MAC address of the active switch determines the stack MAC address.

When the stack initializes, the MAC address of the active switch determines the bridge ID that identifies the stack in the network.

If the active switch changes, the MAC address of the new active switch determines the new bridge ID and stack MAC address.

If the entire switch stack reloads, the switch stack uses the MAC address of the active switch.

Persistent MAC Address on the Switch Stack

You can use the persistent MAC address feature to set a time delay before the stack MAC address changes to the MAC address of the new active stack. When this feature is enabled, the stack MAC address changes in approximately 4 minutes. During this time, if the previous active stack rejoins the stack, the stack continues to use its MAC address as the stack MAC address, even if the switch is now a stack member and not the active stack. If the previous active stack does not rejoin the stack during this period, the switch stack takes the MAC address of the new active stack as the stack MAC address.

You can also configure stack MAC persistency so that the stack MAC address never changes to the new active switch MAC address.

Active and Standby Switch Election and Reelection

All stack members are eligible to be the active stack. If the active stack becomes unavailable, the remaining members elect a new active stack from among themselves.

The active switch is elected or reelected based on one of these factors and in the order listed:

1. The switch that is currently the active switch.
2. The switch with the highest stack member priority value.

**Note**

We recommend assigning the highest priority value to the switch that you prefer to be the active switch ensures that the switch is reelected as active switch if a reelection occurs.

3. The switch that has the configuration file.
4. The switch with the lowest MAC address.

An active stack retains its role unless one of these events occurs:

- The switch stack is reset.*
- The active stack is removed from the switch stack.
- The active stack is reset or powered off.
- The active stack fails.
- The switch stack membership is increased by adding powered-on standalone switches or switch stacks.*

In the events marked by an asterisk (*), the current active stack *might* be reelected based on the listed factors.

When you power on or reset an entire switch stack, some stack members *might not* participate in the active stack election. Stack members that are powered on within the same 20-second time frame participate in the active stack election and have a chance to become the active stack. Stack members that are powered on after the 20-second time frame do not participate in this initial election and become stack members. All stack members participate in reelections. For all powering considerations that affect active-stack elections, see the “Switch Installation” chapter in the hardware installation guide.

The new active stack becomes available after a few seconds. In the meantime, the switch stack uses the forwarding tables in memory to minimize network disruption. The physical interfaces on the other available stack members are not affected during a new active stack election and reset.

After a new active stack is elected and the previous active stack becomes available, the previous active stack *does not* resume its role as the active stack.

For all powering considerations that affect active-stack elections, see the *Catalyst 2960-XR Switch Hardware Installation Guide*.

Switch Stack Configuration Files

The active switch has the saved and running configuration files for the switch stack. All stack members periodically receive synchronized copies of the configuration files from the active switch. If the active switch becomes unavailable, any stack member assuming the role of active switch has the latest configuration files.

The configuration files record these settings:

- System-level (global) configuration settings such as IP, STP, VLAN, and SNMP settings that apply to all stack members
- Stack member interface-specific configuration settings that are specific for each stack member



Note The interface-specific settings of the active switch are saved if the active switch is replaced without saving the running configuration to the startup configuration.

A new, out-of-box device joining a switch stack uses the system-level settings of that switch stack. If a device is moved to a different switch stack before it is powered on, that device loses its saved configuration file and uses the system-level configuration of the new switch stack. If the device is powered on as a standalone device before it joins the new switch stack, the stack will reload. When the stack reloads, the new device may become the device, retain its configuration and overwrite the configuration files of the other stack members.

The interface-specific configuration of each stack member is associated with the stack member number. Stack members retain their numbers unless they are manually changed or they are already used by another member in the same switch stack. If the stack member number changes, the new number goes into effect after that stack member resets.

- If an interface-specific configuration does not exist for that member number, the stack member uses its default interface-specific configuration.
- If an interface-specific configuration exists for that member number, the stack member uses the interface-specific configuration associated with that member number.

If you replace a failed member with an identical model, the replacement member automatically uses the same interface-specific configuration as the failed device. You do not need to reconfigure the interface settings. The replacement device (referred to as the provisioned device) must have the same stack member number as the failed device.

You back up and restore the stack configuration in the same way as you would for a standalone device configuration.

Offline Configuration to Provision a Stack Member

You can use the offline configuration feature to *provision* (to supply a configuration to) a new switch before it joins the switch stack. You can configure the stack member number, the switch type, and the interfaces associated with a switch that is not currently part of the stack. The configuration that you create on the switch stack is called the *provisioned configuration*. The switch that is added to the switch stack and that receives this configuration is called the *provisioned switch*.

You manually create the provisioned configuration through the **switch stack-member-number provision type** global configuration command. You must change the *stack-member-number* on the provisioned switch before you add it to the stack, and it must match the stack member number that you created for the new switch on the switch stack. The switch type in the provisioned configuration must match the switch type of the newly added switch. The provisioned configuration is automatically created when a switch is added to a switch stack and when no provisioned configuration exists.

When you configure the interfaces associated with a provisioned switch, the switch stack accepts the configuration, and the information appears in the running configuration. However, as the switch is not active,

any configuration on the interface is not operational and the interface associated with the provisioned switch does not appear in the display of the specific feature. For example, VLAN configuration information associated with a provisioned switch does not appear in the **show vlan** user EXEC command output on the switch stack.

The switch stack retains the provisioned configuration in the running configuration whether or not the provisioned switch is part of the stack. You can save the provisioned configuration to the startup configuration file by entering the **copy running-config startup-config** privileged EXEC command. The startup configuration file ensures that the switch stack can reload and can use the saved information whether or not the provisioned switch is part of the switch stack.

Effects of Adding a Provisioned Switch to a Switch Stack

When you add a provisioned Device to the switch stack, the stack applies either the provisioned configuration or the default configuration. This table lists the events that occur when the switch stack compares the provisioned configuration with the provisioned switch.

Table 160: Results of Comparing the Provisioned Configuration with the Provisioned Switch

Scenario		Result
The stack member numbers and the Device types match.	<ol style="list-style-type: none"> 1. If the stack member number of the provisioned switch matches the stack member number in the provisioned configuration on the stack, and 2. If the Device type of the provisioned switch matches the Device type in the provisioned configuration on the stack. 	The switch stack applies the provisioned configuration to the provisioned switch and adds it to the stack.
The stack member numbers match but the Device types do not match.	<ol style="list-style-type: none"> 1. If the stack member number of the provisioned switch matches the stack member number in the provisioned configuration on the stack, but 2. The Device type of the provisioned switch does not match the Device type in the provisioned configuration on the stack. 	<p>The switch stack applies the default configuration to the provisioned switch and adds it to the stack.</p> <p>The provisioned configuration is changed to reflect the new information.</p>
The stack member number is not found in the provisioned configuration.		<p>The switch stack applies the default configuration to the provisioned switch and adds it to the stack.</p> <p>The provisioned configuration is changed to reflect the new information.</p>
The stack member number of the provisioned switch is not found in the provisioned configuration.		The switch stack applies the default configuration to the provisioned switch and adds it to the stack.

If you add a provisioned switch that is a different type than specified in the provisioned configuration to a powered-down switch stack and then apply power, the switch stack rejects the (now incorrect) **switch stack-member-number provision type** global configuration command in the startup configuration file. However, during stack initialization, the nondefault interface configuration information in the startup configuration file for the provisioned interfaces (potentially of the wrong type) is executed. Depending on the differences between the actual Device type and the previously provisioned switch type, some commands are rejected, and some commands are accepted.



Note If the switch stack does not contain a provisioned configuration for a new Device, the Device joins the stack with the default interface configuration. The switch stack then adds to its running configuration with a **switch stack-member-number provision type** global configuration command that matches the new Device. For configuration information, see the *Provisioning a New Member for a Switch Stack* section.

Effects of Replacing a Provisioned Switch in a Switch Stack

When a provisioned switch in a switch stack fails, it is removed from the stack, and is replaced with another Device, the stack applies either the provisioned configuration or the default configuration to it. The events that occur when the switch stack compares the provisioned configuration with the provisioned switch are the same as those when you add a provisioned switch to a stack.

Effects of Removing a Provisioned Switch from a Switch Stack

If you remove a provisioned switch from the switch stack, the configuration associated with the removed stack member remains in the running configuration as provisioned information. To completely remove the configuration, use the **no switch stack-member-number provision** global configuration command.

Stack Protocol Version

Each software image includes a *stack protocol version*. The stack protocol version has a *major* version number and a *minor* version number (for example 1.4, where 1 is the major version number and 4 is the minor version number). Both version numbers determine the level of compatibility among the stack members. You can display the stack protocol version by using the **show platform stack manager all** privileged EXEC command.

The switches with the same Cisco IOS software version have the same stack protocol version. Such switches are fully compatible, and all features function properly across the switch stack. A device with the same Cisco IOS software version as the active switch can immediately join the switch stack.

If an incompatibility exists, the fully functional stack members generate a system message that describes the cause of the incompatibility on the specific stack members. The active switch sends the message to all stack members.

For more information, see the *Major Version Number Incompatibility Among Switches* procedure and the *Minor Version Number Incompatibility Among Switches* procedure.

Major Stack Protocol Version Number Incompatibility Among Stack-Capable Switches

Device with different major Cisco IOS software versions usually have different stack protocol versions. Device with different major version numbers are incompatible and cannot exist in the same switch stack.

Minor Stack Protocol Version Number Incompatibility Among Stack-Capable Switches

Switches with the same major version number but with a different minor version number are considered partially compatible. When connected to a switch stack, a partially compatible switch enters version-mismatch (VM) mode and cannot join the stack as a fully functioning member. The software detects the mismatched software and tries to upgrade (or downgrade) the switch in VM mode with the switch stack image or with a tar file image from the switch stack flash memory. The software uses the automatic upgrade (auto-upgrade) and the automatic advise (auto-advise) features.

The port LEDs on switches in version-mismatch mode will also remain off. Pressing the Mode button does not change the LED mode.

Auto-Upgrade

The purpose of the auto-upgrade feature is to allow a switch to be upgraded to a compatible software image, so that the switch can join the switch stack.

When a new switch attempts to join a switch stack, each stack member performs compatibility checks with itself and the new switch. Each stack member sends the results of the compatibility checks to the active stack, which uses the results to determine whether the switch can join the switch stack. If the software on the new switch is incompatible with the switch stack, the new switch enters version-mismatch (VM) mode.

If the auto-upgrade feature is enabled on the existing switch stack, the active stack automatically upgrades the new switch with the same software image running on a compatible stack member. Auto-upgrade starts a few minutes after the mismatched software is detected before starting.

By default, auto-upgrade is enabled (the **boot auto-copy-sw** global configuration command is enabled). You can disable auto-upgrade by using the **no boot auto-copy-sw** global configuration command on the active stack. You can check the status of auto-upgrade by using the **show boot** privileged EXEC command and by checking the *Auto upgrade* line in the display.

Auto-upgrade includes an auto-copy process and an auto-extract process.

- Auto-copy automatically copies the software image running on any stack member to the new switch to automatically upgrade it. Auto-copy occurs if auto-upgrade is enabled, if there is enough flash memory in the new switch, and if the software image running on the switch stack is suitable for the new switch.



Note A switch in VM mode might not run all released software. For example, new switch hardware is not recognized in earlier versions of software.

- Automatic extraction (auto-extract) occurs when the auto-upgrade process cannot find the appropriate software in the stack to copy to the new switch. In that case, the auto-extract process searches all switches in the stack for the tar file needed to upgrade the switch stack or the new switch. The tar file can be in any flash file system in the switch stack or in the new switch. If a tar file suitable for the new switch is found on a stack member, the process extracts the file and automatically upgrades the new switch.

The auto-upgrade (auto-copy and auto-extract) processes start a few minutes after the mismatched software is detected.

When the auto-upgrade process is complete, the new switch reloads and joins the stack as a fully functioning member. If you have both stack cables connected during the reload, network downtime does not occur because the switch stack operates on two rings.

Auto-Advise

Automatic advise (auto-advise) occurs when the auto-upgrade process cannot find appropriate stack member software to copy to the new switch. This process tells you the command (**archive copy-sw** or **archive download-sw** privileged EXEC command) and the image name (tar filename) needed to manually upgrade the switch stack or the new switch. The recommended image can be the running switch stack image or a tar file in any flash file system in the switch stack (including the new switch). If an appropriate image is not found in the stack flash file systems, the auto-advise process tells you to install new software on the switch stack. Auto-advise cannot be disabled, and there is no command to check its status.

Examples of Auto-Advise Messages

When you add a switch that has a different minor version number to the switch stack, the software displays messages in sequence (assuming that there are no other system messages generated by the switch).

This example shows that the switch stack detected a new switch that is running a different minor version number than the switch stack. Auto-copy starts, finds suitable software to copy from a stack member to the switch in VM mode, upgrades the switch in VM mode, and then reloads it:

```
*Mar 11 20:31:19.247:%STACKMGR-6-STACK_LINK_CHANGE:Stack Port 2 Switch 2 has changed to
state UP
*Mar 11 20:31:23.232:%STACKMGR-6-SWITCH_ADDED_VM:Switch 1 has been ADDED to the
stack(VERSION_MISMATCH)
*Mar 11 20:31:23.291:%STACKMGR-6-SWITCH_ADDED_VM:Switch 1 has been ADDED to the
stack(VERSION_MISMATCH) (Stack 1-3)
*Mar 11 20:33:23.248:%IMAGEMGR-6-AUTO_COPY_SW_INITIATED:Auto-copy-software process initiated
for switch number(s) 1
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:Searching for stack member to act
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:as software donor...
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:Found donor (system #2) for
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:member(s) 1
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:System software to be uploaded:
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:System Type: 0x00000000
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:archiving c2960x-universalk9-mz.150-2.EX1
(directory)
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:archiving c2960x-universalk9-mz.150-2.EX1.bin
(4945851 bytes)
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:archiving
c2960x-universalk9-mz.150-2.EX1/info(450 bytes)
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:archiving info (104 bytes)
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:examining image...
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:extracting info (104 bytes)
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:extracting
c2960x-universalk9-mz.150-2.EX1/info(450 bytes)
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:extracting info (104 bytes)
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:Stacking Version Number:1.4
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:System Type: 0x00000000
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW: Ios Image File Size: 0x004BA200
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW: Total Image File Size:0x00818A00
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW: Minimum Dram required:0x08000000
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW: Image Suffix:universalk9-mz.150-2.EX1
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW: Image Directory:c2960x-universalk9-mz.150-2.EX1
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW: Image Name:c2960x-universalk9-mz.150-2.EX1
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW: Image 1:flash1:c2960x-universalk9-mz.150-2.EX1
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW: Old image will be deleted after download.
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:Extracting images from archive into flash on
switch 1...
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:c2960x-universalk9-mz.150-2.EX1 (directory)
```

```

*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:extracting
c2960x-universalk9-mz.150-2.EX1/c2960x-universalk9-mz.150-2.EX1 (4945851 bytes)
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:extracting c2960x-universalk9-mz.150-2.EX1/info
(450 bytes)
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:extracting info (104 bytes)
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:Installing
(renaming):`flash1:update/c2960x-universalk9-mz.150-2.EX1' ->
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW: `flash1:c2960x-universalk9-mz.150-2.EX1'
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:New software image installed in
flash1:c2960x-universalk9-mz.150-2.EX1
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:Removing old
image:flash1:c2960x-universalk9-mz.150-2.EX1
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:All software images installed.
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:Requested system reload in progress...
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:Software successfully copied to
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:system(s) 1
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:Done copying software
*Mar 11 20:36:15.038:%IMAGEMGR-6-AUTO_COPY_SW:Reloading system(s) 1

```

This example shows that the switch stack detected a new switch that is running a different minor version number than the switch stack. Auto-copy starts but cannot find software in the switch stack to copy to the VM-mode switch to make it compatible with the switch stack. The auto-advise process starts and recommends that you download a tar file from the network to the switch in VM mode:

```

*Mar 1 00:01:11.319:%STACKMGR-6-STACK_LINK_CHANGE:Stack Port 2 Switch 2 has changed to state
UP
*Mar 1 00:01:15.547:%STACKMGR-6-SWITCH_ADDED_VM:Switch 1 has been ADDED to the stack
(VERSION_MISMATCH)
stack_2#
*Mar 1 00:03:15.554:%IMAGEMGR-6-AUTO_COPY_SW_INITIATED:Auto-copy-software process initiated
for switch number(s) 1
*Mar 1 00:03:15.554:%IMAGEMGR-6-AUTO_COPY_SW:
*Mar 1 00:03:15.554:%IMAGEMGR-6-AUTO_COPY_SW:Searching for stack member to act
*Mar 1 00:03:15.554:%IMAGEMGR-6-AUTO_COPY_SW:as software donor...
*Mar 1 00:03:15.554:%IMAGEMGR-6-AUTO_COPY_SW:Software was not copied
*Mar 1 00:03:15.562:%IMAGEMGR-6-AUTO_ADVISE_SW_INITIATED:Auto-advise-software process
initiated for switch number(s) 1
*Mar 1 00:04:22.537:%IMAGEMGR-6-AUTO_ADVISE_SW:
*Mar 1 00:04:22.537:%IMAGEMGR-6-AUTO_ADVISE_SW:
*Mar 1 00:04:22.537:%IMAGEMGR-6-AUTO_ADVISE_SW:Systems with incompatible software
*Mar 1 00:04:22.537:%IMAGEMGR-6-AUTO_ADVISE_SW:have been added to the stack. The
*Mar 1 00:04:22.537:%IMAGEMGR-6-AUTO_ADVISE_SW:storage devices on all of the stack
*Mar 1 00:04:22.537:%IMAGEMGR-6-AUTO_ADVISE_SW:members have been scanned, and it has
*Mar 1 00:04:22.537:%IMAGEMGR-6-AUTO_ADVISE_SW:been determined that the stack can be
*Mar 1 00:04:22.537:%IMAGEMGR-6-AUTO_ADVISE_SW:repaired by issuing the following
*Mar 1 00:04:22.537:%IMAGEMGR-6-AUTO_ADVISE_SW:command(s) :
*Mar 1 00:04:22.537:%IMAGEMGR-6-AUTO_ADVISE_SW:
*Mar 1 00:04:22.537:%IMAGEMGR-6-AUTO_ADVISE_SW: archive download-sw /force-reload /overwrite
/dest 1 flash1:c2960x-universalk9-mz.150-2.EX1.tar
*Mar 1 00:04:22.537:%IMAGEMGR-6-AUTO_ADVISE_SW:

```



Note Auto-advise and auto-copy identify which images are running by examining the info file and by searching the directory structure on the switch stack. If you download your image by using the **copy tftp:** boot loader command instead of the **archive download-sw** privileged EXEC command, the proper directory structure is not created.

SDM Template Mismatch in Switch Stacks

All stack members use the Switch Database Management (SDM) template configured on the active switch. When a new switch is added to a stack, the SDM configuration that is stored on the active switch overrides the template configured on an individual switch.

You can use the **show switch** privileged EXEC command to see if any stack members are in SDM-mismatch mode.

Version-mismatch (VM) mode has priority over SDM-mismatch mode. If a VM-mode condition and an SDM-mismatch mode exist, the switch stack first attempts to resolve the VM-mode condition.

For more information about SDM templates, see the *Catalyst 2960-XR Switch System Management Configuration Guide*.

Switch Stack Management Connectivity

You manage the switch stack and the stack member interfaces through the active switch. You can use the CLI, SNMP, and any of the supported network management applications. You cannot manage stack members on an individual device basis.

Connectivity to Specific Stack Members

If you want to configure a specific stack member port, you must include the stack member number in the CLI command interface notation.

To debug a specific stack member, you can access it from the active stack by using the **session stack-member-number** privileged EXEC command. The stack member number is appended to the system prompt. For example, Switch-2# is the prompt in privileged EXEC mode for stack member 2, and the system prompt for the active stack is Switch. Only the **show** and **debug** commands are available in a CLI session to a specific stack member.

Connectivity to the Switch Stack Through an IP Address

The switch stack is managed through a single IP address. The IP address is a system-level setting and is not specific to the active stack or to any other stack member. You can still manage the stack through the same IP address even if you remove the active stack or any other stack member from the stack, provided there is IP connectivity.



Note Stack members retain their IP addresses when you remove them from a switch stack. To avoid a conflict by having two devices with the same IP address in your network, change the IP addresses of any active stack that you remove from the switch stack.

For related information about switch stack configurations, see the *Switch Stack Configuration Files* section.

Connectivity to the Switch Stack Through Console Ports or Ethernet Management Ports

You can connect to the active switch by using one of these methods:

- You can connect a terminal or a PC to the active switch through the console port of one or more stack members.
- You can connect a PC to the active switch through the Ethernet management ports of one or more stack members. For more information about connecting to the switch stack through Ethernet management ports, see the *Using the Ethernet Management Port* section.

You can connect to the active switch by connecting a terminal or a PC to the active switch through the console port of one or more stack members.

When you use the console port of a stack member, a VTY session is created with the IP address in the 192.168.0.1/24 subnet.

Be careful when using multiple CLI sessions to the active switch. Commands that you enter in one session are not displayed in the other sessions. Therefore, it is possible that you might not be able to identify the session from which you entered a command.

We recommend using only one CLI session when managing the switch stack.

How to Configure a Switch Stack

Enabling the Persistent MAC Address Feature



Note When you enter the command to configure this feature, a warning message appears with the consequences of your configuration. You should use this feature cautiously. Using the old active switch MAC address elsewhere in the same domain could result in lost traffic.

Follow these steps to enable persistent MAC address:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **stack-mac persistent timer [0 | *time-value*]**
4. **end**
5. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable	Enables privileged EXEC mode.

	Command or Action	Purpose
	<p>Example:</p> <pre>Device> enable</pre>	<ul style="list-style-type: none"> Enter your password if prompted.
Step 2	<p>configure terminal</p> <p>Example:</p> <pre>Device# configure terminal</pre>	Enters global configuration mode.
Step 3	<p>stack-mac persistent timer [0 <i>time-value</i>]</p> <p>Example:</p> <pre>Device(config)# stack-mac persistent timer 7</pre>	<p>Enables a time delay after a stack-active switch change before the stack MAC address changes to that of the new ac. If the previous active switch rejoins the stack during this period, the stack uses that MAC address as the stack MAC address.</p> <p>You can configure the time period as 0 to 60 minutes.</p> <ul style="list-style-type: none"> Enter the command with no value to set the default delay of approximately 4 minutes. We recommend that you always enter a value. <p>If the command is entered without a value, the time delay appears in the running-config file with an explicit timer value of 4 minutes.</p> <ul style="list-style-type: none"> Enter 0 to continue using the MAC address of the current active switch indefinitely. <p>The stack MAC address of the previous active switch is used until you enter the no stack-mac persistent timer command, which immediately changes the stack MAC address to that of the current active switch.</p> <ul style="list-style-type: none"> Enter a <i>time-value</i> from 1 to 60 minutes to configure the time period before the stack MAC address changes to the new active switch. <p>The stack MAC address of the previous active switch is used until the configured time period expires or until you enter the no stack-mac persistent timer command.</p> <p>Note If you enter the no stack-mac persistent timer command after a new active switch takes over, before the time expires, the switch stack moves to the current active switch MAC address.</p>
Step 4	<p>end</p> <p>Example:</p>	Returns to privileged EXEC mode.

	Command or Action	Purpose
	Device(config)# end	
Step 5	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

What to do next

Use the **no stack-mac persistent timer** global configuration command to disable the persistent MAC address feature.

Assigning a Stack Member Number

This optional task is available only from the active stack.

Follow these steps to assign a member number to a stack member:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **switch** *current-stack-member-number* **renumber** *new-stack-member-number*
4. **end**
5. **reload slot** *stack-member-number*
6. **show switch**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.

	Command or Action	Purpose
Step 3	switch <i>current-stack-member-number</i> renumber <i>new-stack-member-number</i> Example: Device(config)# switch 3 renumber 4	Specifies the current stack member number and the new stack member number for the stack member. The range is 1 to 8. You can display the current stack member number by using the show switch user EXEC command.
Step 4	end Example: Device(config)# end	Returns to privileged EXEC mode.
Step 5	reload slot <i>stack-member-number</i> Example: Device# reload slot 4	Resets the stack member.
Step 6	show switch Example: showDevice	Verify the stack member number.
Step 7	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Setting the Stack Member Priority Value

This optional task is available only from the active stack.

Follow these steps to assign a priority value to a stack member:

SUMMARY STEPS

1. **enable**
2. **switch** *stack-member-number* **priority** *new-priority-number*
3. **show switch** *stack-member-number*
4. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example:	Enables privileged EXEC mode. Enter your password if prompted.

	Command or Action	Purpose
	Device enable	
Step 2	switch <i>stack-member-number</i> priority <i>new-priority-number</i> Example: Device# switch 3 priority 2	Specifies the stack member number and the new priority for the stack member. The stack member number range is 1 to 8. The priority value range is 1 to 15. You can display the current priority value by using the show switch user EXEC command. The new priority value takes effect immediately but does not affect the current active stack. The new priority value helps determine which stack member is elected as the new active stack when the current active stack or switch stack resets.
Step 3	show switch <i>stack-member-number</i> Example: Device# show switch	Verify the stack member priority value.
Step 4	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Provisioning a New Member for a Switch Stack

This optional task is available only from the active switch.

SUMMARY STEPS

1. **show switch**
2. **configure terminal**
3. **switch** *stack-member-number* **provision** *type*
4. **end**
5. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	show switch Example: Device# show switch	Displays summary information about the switch stack.
Step 2	configure terminal Example:	Enters global configuration mode.

	Command or Action	Purpose
	Device# <code>configure terminal</code>	
Step 3	switch <i>stack-member-number</i> provision <i>type</i> Example: Device(config)# <code>switch 3 provision WS-xxxx</code>	Specifies the stack member number for the preconfigured switch. By default, no switches are provisioned. For <i>stack-member-number</i> , the range is 1 to 8. Specify a stack member number that is not already used in the switch stack. See Step 1. For <i>type</i> , enter the model number of a supported switch that is listed in the command-line help strings.
Step 4	end Example: Device(config)# <code>end</code>	Returns to privileged EXEC mode.
Step 5	copy running-config startup-config Example: Device# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Removing Provisioned Switch Information

Before you begin, you must remove the provisioned switch from the stack. This optional task is available only from the active stack.

SUMMARY STEPS

1. `configure terminal`
2. `no switch stack-member-number provision`
3. `end`
4. `copy running-config startup-config`

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode.
Step 2	no switch <i>stack-member-number</i> provision Example: Device(config)# <code>no switch 3 provision</code>	Removes the provisioning information for the specified member.

	Command or Action	Purpose
Step 3	end Example: Device(config)# end	Returns to privileged EXEC mode.
Step 4	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Example

If you are removing a provisioned switch in a stack with this configuration:

- The stack has four members
- Stack member 1 is the active stack
- Stack member 3 is a provisioned switch

and want to remove the provisioned information and to avoid receiving an error message, you can remove power from stack member 3, disconnect the stack cables between the stack member 3 and switches to which it is connected, reconnect the cables between the remaining stack members, and enter the **no switch stack-member-number provision** global configuration command.

Troubleshooting the Switch Stack

Accessing the CLI of a Specific Member

This optional task is for debugging purposes, and is available only from the active switch.

You can access all or specific members by using the **remote command** {**all** | *stack-member-number*} privileged EXEC command. The stack member number range is 1 to 8.

You can access specific members by using the **session** *stack-member-number* privileged EXEC command. The member number is appended to the system prompt. For example, the prompt for member 2 is **Switch-2#**, and system prompt for the active switch is **Switch#**. Enter **exit** to return to the CLI session on the active switch. Only the **show** and **debug** commands are available on a specific member.

Temporarily Disabling a Stack Port

If a stack port is flapping and causing instability in the stack ring, to disable the port, enter the **switch stack-member-number stack port port-number disable** privileged EXEC command. To reenble the port, enter the **switch stack-member-number stack port port-number enable** command.



Note Be careful when using the **switch stack-member-number stack port port-number disable** command. When you disable the stack port, the stack operates at half bandwidth.

A stack is in the full-ring state when all members are connected through the stack ports and are in the ready state.

The stack is in the partial-ring state when the following occurs:

- All members are connected through their stack ports but some are not in the ready state.
- Some members are not connected through the stack ports.

SUMMARY STEPS

1. **switch stack-member-number stack port port-number disable**
2. **switch stack-member-number stack port port-number enable**

DETAILED STEPS

	Command or Action	Purpose
Step 1	switch stack-member-number stack port port-number disable Example: Device# switch 2 stack port 1 disable	Disables the specified stack port.
Step 2	switch stack-member-number stack port port-number enable Example: Device# switch 2 stack port 1 enable	Reenables the stack port.

When you disable a stack port and the stack is in the full-ring state, you can disable only one stack port. This message appears:

```
Enabling/disabling a stack port may cause undesired stack changes. Continue?[confirm]
```

When you disable a stack port and the stack is in the partial-ring state, you cannot disable the port. This message appears:

```
Disabling stack port not allowed with current stack configuration.
```

Reenabling a Stack Port While Another Member Starts

Stack Port 1 on Switch 1 is connected to Port 2 on Switch 4. If Port 1 is flapping, you can disable Port 1 with the **switch 1 stack port 1 disable** privileged EXEC command. While Port 1 on Switch 1 is disabled and Switch 1 is still powered on, follow these steps to reenabling a stack port:

-
- Step 1** Disconnect the stack cable between Port 1 on Switch 1 and Port 2 on Switch 4.
- Step 2** Remove Switch 4 from the stack.
- Step 3** Add a switch to replace Switch 4 and assign it switch-number 4.
- Step 4** Reconnect the cable between Port 1 on Switch 1 and Port 2 on Switch 4 (the replacement switch).
- Step 5** Reenable the link between the switches. Enter the **switch 1 stack port 1 enable** privileged EXEC command to enable Port 1 on Switch 1.
- Step 6** Power on Switch 4.
-



Caution Powering on Switch 4 before enabling the Port 1 on Switch 1 might cause one of the switches to reload. If Switch 4 is powered on first, you might need to enter the **switch 1 stack port 1 enable** and the **switch 4 stack port 2 enable** privileged EXEC commands to bring up the link.

Monitoring the Device Stack

Table 161: Commands for Displaying Stack Information

Command	Description
show controller ethernet-controller stack port {1 2}	Displays stack port counters (or per-interface and per-stack port send and receive statistics read from the hardware).
show controller ethernet-controller fastethernet0	Displays information about the Ethernet management port, including the port status and the per-interface send and receive statistics read from the hardware.
show platform stack compatibility	Displays information about HULC feature compatibility.
show platform stack manager all	Displays all stack manager information, such as the stack protocol version.
show platform stack passive-links	Displays information about stack passive links.
show switch	Displays summary information about the stack, including the status of provisioned switches and switches in version-mismatch mode.
show switch <i>stack-member-number</i>	Displays information about a specific member.
show switch detail	Displays detailed information about the stack.
show switch neighbors	Displays the stack neighbors.
show switch stack-ports	Displays port information for the stack.

Configuration Examples for Switch Stacks

Switch Stack Configuration Scenarios

Most of these switch stack configuration scenarios assume that at least two device are connected through their stack ports.

Table 162: Configuration Scenarios

Scenario		Result
Active switch election specifically determined by existing active switches	Connect two powered-on switch stacks through the stack ports.	Only one of the two active switches becomes the new active switch.
Active switch election specifically determined by the stack member priority value	<ol style="list-style-type: none"> 1. Connect two switches through their stack ports. 2. Use the switch stack-member-number priority new-priority-number global configuration command to set one stack member with a higher member priority value. 3. Restart both stack members at the same time. 	The stack member with the higher priority value is elected active switch.
Active switch election specifically determined by the configuration file	<p>Assuming that both stack members have the same priority value:</p> <ol style="list-style-type: none"> 1. Make sure that one stack member has a default configuration and that the other stack member has a saved (nondefault) configuration file. 2. Restart both stack members at the same time. 	The stack member with the saved configuration file is elected active switch.
Active switch election specifically determined by the MAC address	Assuming that both stack members have the same priority value, configuration file, and feature set, restart both stack members at the same time.	The stack member with the lower MAC address is elected active switch.

Scenario		Result
Stack member number conflict	<p>Assuming that one stack member has a higher priority value than the other stack member:</p> <ol style="list-style-type: none"> 1. Ensure that both stack members have the same stack member number. If necessary, use the switch <i>current-stack-member-number</i> renumber <i>new-stack-member-number</i> global configuration command. 2. Restart both stack members at the same time. 	The stack member with the higher priority value retains its stack member number. The other stack member has a new stack member number.
Add a stack member	<ol style="list-style-type: none"> 1. Power off the new switch. 2. Through their stack ports, connect the new switch to a powered-on switch stack. 3. Power on the new switch. 	The active switch is retained. The new switch is added to the switch stack.
Active switch failure	Remove (or power off) the active switch.	One of the remaining stack members becomes the new active switch. All other stack members in the stack remain as stack members and do not reboot.
Add more than eight stack members	<ol style="list-style-type: none"> 1. Through their stack ports, connect nine device. 2. Power on all device. 	<p>Two device become active switches. One active switch has eight stack members. The other active switch remains as a standalone device.</p> <p>Use the Mode button and port LEDs on the device to identify which device are active switches and which device belong to each active switch.</p>

Enabling the Persistent MAC Address Feature: Example

This example shows how to configure the persistent MAC address feature for a 7-minute time delay and to verify the configuration:

```

Device(config)# stack-mac persistent timer 7
WARNING: The stack continues to use the base MAC of the old Master
WARNING: as the stack MAC after a master switchover until the MAC
WARNING: persistency timer expires. During this time the Network
WARNING: Administrators must make sure that the old stack-mac does
WARNING: not appear elsewhere in this network domain. If it does,
WARNING: user traffic may be blackholed.
Device(config)# end
Device# show switch

```

```
Switch/Stack Mac Address : 0016.4727.a900
Mac persistency wait time: 7 mins

Switch#  Role   Mac Address      Priority  H/W   Current
-----  -
*1       Master 0016.4727.a900   1        P2B   Ready
```

Provisioning a New Member for a Switch Stack: Example

This example shows how to provision a switch with a stack member number of 2 for the switch stack. The **show running-config** command output shows the interfaces associated with the provisioned switch:

```
Device(config)# switch 2 provision switch_PID
Device(config)# end
Device# show running-config | include switch 2
switch 2 provision switch_PID
```

Additional References for Switch Stacks

Related Documents

Related Topic	Document Title
Cabling and powering on a switch stack.	<i>Catalyst 2960-XR Switch Hardware Installation Guide</i> http://www.cisco.com/c/en/us/td/docs/switches/lan/catalyst2960cx_3650cx/hardware/installation/guide/b_2960cx-3560cx_hig.html

Error Message Decoder

Description	Link
To help you research and resolve system error messages in this release, use the Error Message Decoder tool.	https://www.cisco.com/cgi-bin/Support/Errordecoder/index.cgi

Standards and RFCs

Standard/RFC	Title
None	—

MIBs

MIB	MIBs Link
All the supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and software images, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
<p>The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies.</p> <p>To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds.</p> <p>Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.</p>	<p>http://www.cisco.com/support</p>



CHAPTER 83

FlexStack-Extended

This module describes the FlexStack-Extended feature supported on Catalyst 2960-X Series Switches with LAN Base license and Cisco Catalyst 2960-XR Series Switches.

- [Restrictions for FlexStack-Extended, on page 1665](#)
- [Information About FlexStack-Extended, on page 1665](#)
- [How to Configure FlexStack-Extended, on page 1668](#)
- [Configuration Examples for FlexStack-Extended, on page 1671](#)
- [Feature Information for FlexStack-Extended, on page 1672](#)

Restrictions for FlexStack-Extended

The following restrictions apply to the horizontal stacking of switches.

- For fiber module, both ports must be configured as either network ports or stack ports. Do not configure one port as a network port and the other as a stack port.
- Online Insertion and Removal (OIR) is possible only with the same type of port.
- To connect a switch with a FlexStack module to a switch with a hybrid module, set the speed manually. The stack speed should be set to 10G.
- If the stack module (hybrid or fiber) on a switch is replaced with a FlexStack module, the bandwidth must to be reset manually.

Information About FlexStack-Extended

FlexStack-Extended

Prior to Cisco IOS Release 15.2(6)E, stacking was supported with FlexStack-Plus module, which has two copper stack ports. Copper stack ports support short reach connectivity across local switches. FlexStack-Extended overcomes the problem of short reach connectivity by using 10G SFP+ ports to enable stacking that allows long reach stacking using optics.

The same models that support FlexStack-Plus on Cisco Catalyst 2960-X Series Switches and Cisco Catalyst 2960-XR Series Switches support FlexStack-Extended.

When you convert a network port to a stack port, it continues to work as a network port without any impact to the current running configuration until the next reload of the switch.

When you convert a stack port back to a network port, it continues to work as a stack port until the next reload. After reload, the port comes up as a network port with the default configuration.



Note When uplink ports are working as stack ports, these particular uplink interfaces (for example, TenGigabitEthernet 1/1/1) are not displayed in any show command or are not available under any configuration command, unlike other network ports. These uplink interfaces are made available only after the reload of the switch; once ports are converted back to network ports.

FlexStack-Extended on Catalyst 2960-X and 2960-XR Switches

Cisco Catalyst 2960-X and 2960-XR Series Switches support FlexStack-Extended with hybrid stack and fiber stack modules, and also with 10G SFP+ front panel uplink ports.

The following models support FlexStack-Extended with hybrid stack and fiber stack modules:

- Cisco Catalyst 2960X-24PD-L
- Cisco Catalyst 2960X-24PS-L
- Cisco Catalyst 2960X-24TD-L
- Cisco Catalyst 2960X-24TS-L
- Cisco Catalyst 2960X-48FPD-L
- Cisco Catalyst 2960X-48FPS-L
- Cisco Catalyst 2960X-48LPD-L
- Cisco Catalyst 2960X-48LPS-L
- Cisco Catalyst 2960X-48TD-L
- Cisco Catalyst 2960XR-24PD-I
- Cisco Catalyst 2960XR-24PS-I
- Cisco Catalyst 2960XR-24TD-I
- Cisco Catalyst 2960XR-48FPD-I
- Cisco Catalyst 2960XR-48FPS-I
- Cisco Catalyst 2960XR-48LPD-I
- Cisco Catalyst 2960XR-48LPS-I
- Cisco Catalyst 2960XR-48TD-I

The following models support front-panel stacking:

- Cisco Catalyst 2960X-24TD-L
- Cisco Catalyst 2960X-48FPD-L

- Cisco Catalyst 2960X-48LPD-L
- Cisco Catalyst 2960X-48TD-L
- Cisco Catalyst 2960XR-24PD-I
- Cisco Catalyst 2960XR-24TD-I
- Cisco Catalyst 2960XR-48FPD-I
- Cisco Catalyst 2960XR-48LPD-I
- Cisco Catalyst 2960XR-48TD-I

A hybrid stack module has one copper stack connector and one SFP+ port. The copper port allows short-reach connectivity across the local stack of switches, and the SFP+ allows for long-reach stacking using standard optics. Hybrid-stack module ports can only be used as stack ports. The SFP+ port of the module cannot be changed to a network port.

A fiber stack module has two SFP+ interfaces, which allows for long-reach stacking using standard optics. Fiber stack ports are used either as network ports or stack ports. By default all ports on the fiber stack module are stack port. These ports can be converted to network ports.



Note Stack fast convergence is not supported on hybrid stack and fiber stack modules.

The stack bandwidth for the following stack configuration is 40G:

- Stack using hybrid stack module.
- Stack using fiber stack module.
- Stack using FlexStack-Plus, hybrid, and fiber modules.

For more information on Installing the Switch, see the *Catalyst 2960-X and 2960-XR Switch Hardware Installation Guide* on www.cisco.com.

In Cisco IOS Release 15.2(6)E, FlexStack-Extended is supported on C2960X-HYBRID-STK and C2960X-FIBER-STK modules. By default, all ports are treated as stack ports. Online Insertion and Removal (OIR) of these module (hot swappable) is supported; however, these should be replaced with the same module type. If the module is replaced by a different module type, a reload is required.

The SFP+ port in C2960X-HYBRID-STK module can only be used as a stack port.

Two 10G SFP+ stack ports in the C2960X-FIBER-STK module can be converted to network ports. Using one port as uplink and the other as a stack port is not supported. The SFP+ ports are displayed as Te1/1/1 and Te1/1/2 when converted to network ports.

All SFP+ optics supported by front panel uplink ports are supported by these modules.

Default Port Configurations

The following section lists the default port configurations:

Hybrid Stack

- The default is stack port.

Fiber Stack

- The default is stack port.

FlexStack-Extended LED

The light-emitting diode (LED) behavior is the same for stack ports and network ports. The LED status is as given below:

- OFF—Cable removed/no cable/the switch is off.
- Solid green—Cable inserted and link is up.
- Blinking green—Traffic is running.
- Blinking amber—Cable is connected and the link is coming up.

How to Configure FlexStack-Extended

The 10G SFP+ can be used either as a network port or a stack port.

- All TenGigabitEthernet ports available on the active stack and all stack members that are capable of FlexStack-Extended can be converted to network ports or stack ports.
- If any TenGigabitEthernet port is converted to a horizontal stack port, the stack port number (1 or 2) is displayed corresponding to that port.



Note You cannot choose one stack port from the front panel and another from the back panel. Both stack ports should either be from the front panel or back panel. The following example shows how to configure hstack ports:

```
switch 1 hstack-port 1 Tengigabitethernet 1/0/1
switch 1 hstack-port 2 Tengigabitethernet 1/0/2
```

Configuring a Stack Port as a Network Port

You can configure both 10G stack ports as network ports.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **no switch** *switch-number* **hstack-port** *stack-port*
4. **exit**
5. **reload**
6. **show switch hstack-ports**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables Privileged EXEC mode. <ul style="list-style-type: none">• Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	no switch switch-number hstack-port stack-port Example: Device(config)# no switch 1 hstack-port 1	Configures the stack port as a network port. <ul style="list-style-type: none">• The TenGigabitEthernet interface number is automatically added when the command is configured.
Step 4	exit Example: Device(config)# exit	Exits global configuration mode and returns to privileged EXEC mode.
Step 5	reload Example: Device# reload	Reloads a device. <ul style="list-style-type: none">• Save the configuration by using the copy running-config startup-config command before reloading a device.
Step 6	show switch hstack-ports Example: Device# show switch hstack-ports	Shows the current status and the next reload status for ports. Note For Flexstack-Plus and Hybrid stack modules, port numbers are not displayed.

What to do next

The following is sample output from the **show switch hstack-ports** command:

```
Device# show switch hstack-ports

Horizontal stack port status :
Te Ports  Stack Port  Operational Status  Next Reload Status  Media Type
-----
Tel/0/1   NA           N/W Port           N/W Port           Fiber
Tel/0/2   NA           N/W Port           N/W Port           Fiber
Tel/1/1   NA           N/W port           N/W port           Fiber
Tel/1/2   NA           N/W Port           N/W port           Fiber
```

Configuring a Network Port as a Stack Port

You can configure both 10G Network ports as stack ports.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **switch** *switch-number* **hstack-port** *stack-port* *interface-id* [**tengigabitethernet** *interface-number*]
4. **exit**
5. **reload**
6. **show switch hstack-ports**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables Privileged EXEC mode. <ul style="list-style-type: none">• Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	switch <i>switch-number</i> hstack-port <i>stack-port</i> <i>interface-id</i> [tengigabitethernet <i>interface-number</i>] Example: Device(config)# switch 1 hstack-port 1 Tengigabitethernet 1/1/1	Configures the network port as a stack port.
Step 4	exit Example: Device(config)# exit	Exits global configuration mode and returns to privileged EXEC mode.
Step 5	reload Example: Device# reload	Reloads a device. <ul style="list-style-type: none">• Save the configuration by using the copy running-config startup-config command before reloading a device.
Step 6	show switch hstack-ports Example: Device# show switch hstack-ports	Shows the current status and the next reload status for the ports. Note For Flexstack-Plus and Hybrid stack modules, the port numbers cannot be seen.

Example

The following is sample output from the **show switch hstack-ports** command:

```
Device# show switch hstack-ports
```



```

Horizontal stack port status :
-----
Te Ports  Stack Port  Operational Status  Next Reload Status  Media Type
-----
Tel/0/1   NA           N/W Port           N/W Port           Fiber
Tel/0/2   NA           N/W Port           N/W Port           Fiber
Tel/1/1   1           Stack Port         Stack Port         Fiber
Tel/1/2   2           Stack Port         Stack Port         Fiber

```

Configuring the Stack Speed

The speed change is configured on the back stack port with a FlexStack-Plus module. Perform this task to configure the stack speed.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **switch stack port-speed *speed***
4. **exit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables Privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	switch stack port-speed <i>speed</i> Example: Device(config)# switch stack port-speed 10	Configures the speed of the switch stack port. Note Use the no form of the command to change the stack speed.
Step 4	exit Example: Device(config)# exit	Exits global configuration mode and returns to privileged EXEC mode.

Configuration Examples for FlexStack-Extended

Examples: Configuring FlexStack-Extended

The following example shows how to convert a stack port to network port:

```
Device> enable
Device# configure terminal
Device(config)# no switch 1 hstack-port 1

Do you want to continue?[confirm]
New port setting will be effective after next reload
```

The following is sample output from the **show switch hstack-ports** command:

```
Device# show switch hstack-ports

Horizontal stack port status :
-----
Te Ports  Stack Port  Operational Status  Next Reload Status  Media Type
-----
Tel1/0/1  NA              N/W Port            N/W Port            Fiber
Tel1/0/2  NA              N/W Port            N/W Port            Fiber
Tel1/1/1  NA              N/W Port            N/W Port            Fiber
Tel1/1/2  NA              N/W Port            N/W Port            Fiber
```

The following example shows how you can set the speed of the switch stack port:

```
Device> enable
Device# configure terminal
Device(config)# switch stack port-speed 10
Device(config)# end
```

Feature Information for FlexStack-Extended

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Table 163: Feature Information for FlexStack-Extended

Feature Name	Release	Feature Information
FlexStack-Extended	Cisco IOS Release 15.2(6)E	Switches that support 10G Small Form-Factor Pluggable (SFP+) uplink ports can be part of horizontal stacking. Based on your requirement, create a half-ring or a full-ring stack, and remaining uplink ports can continue to work as network ports. In Cisco IOS Release 15.2(6)E, this feature was implemented on the following platforms: <ul style="list-style-type: none"> • Cisco Catalyst 2960-XR Series Switches



PART **XIV**

VLAN

- [Configuring VTP, on page 1675](#)
- [VLANs, on page 1699](#)
- [Configuring VLAN Trunks, on page 1719](#)
- [Configuring Private VLANs, on page 1739](#)
- [Configuring VMPS, on page 1761](#)
- [Configuring IEEE 802.1Q and Layer 2 Protocol Tunneling, on page 1775](#)
- [Configuring Voice VLANs, on page 1801](#)



CHAPTER 84

Configuring VTP

- [Prerequisites for VTP, on page 1675](#)
- [Information About VTP, on page 1676](#)
- [Default VTP Configuration, on page 1683](#)
- [How to Configure VTP, on page 1684](#)
- [Monitoring VTP, on page 1694](#)
- [Configuration Examples for VTP, on page 1695](#)
- [Where to Go Next, on page 1696](#)
- [Additional References, on page 1697](#)
- [Feature History and Information for VTP, on page 1697](#)

Prerequisites for VTP

The following are prerequisites for VTP:

- Before you create VLANs, you must decide whether to use the VLAN Trunking Protocol (VTP) in your network. Using VTP, you can make configuration changes centrally on one or more switches and have those changes automatically communicated to all the other switches in the network. Without VTP, you cannot send information about VLANs to other switches. VTP is designed to work in an environment where updates are made on a single switch and are sent through VTP to other switches in the domain. It does not work well in a situation where multiple updates to the VLAN database occur simultaneously on switches in the same domain, which would result in an inconsistency in the VLAN database.
- The switch supports 1005 VLANs when running the IP Lite image.
- The switch supports 255 VLANs when running the LAN base feature set.
- However, the number of routed ports, SVIs, and other configured features affects the usage of the switch hardware. If the switch is notified by VTP of a new VLAN and the switch is already using the maximum available hardware resources, it sends a message that there are not enough hardware resources available and shuts down the VLAN. The output of the **show vlan user EXEC** command shows the VLAN in a suspended state.

Information About VTP

VTP

VTP is a Layer 2 messaging protocol that maintains VLAN configuration consistency by managing the addition, deletion, and renaming of VLANs on a network-wide basis. VTP minimizes misconfigurations and configuration inconsistencies that can cause several problems, such as duplicate VLAN names, incorrect VLAN-type specifications, and security violations.

VTP version 1 and version 2 support only normal-range VLANs (VLAN IDs 1 to 1005). VTP version 3 supports the entire VLAN range (VLANs 1 to 4094). Extended range VLANs (VLANs 1006 to 4094) are supported only in VTP version 3.

You cannot convert from VTP version 3 to VTP version 2 if extended VLANs are configured in the domain.

VTP Domain

A VTP domain (also called a VLAN management domain) consists of one device or several interconnected devices or device stacks under the same administrative responsibility sharing the same VTP domain name. A device can be in only one VTP domain. You make global VLAN configuration changes for the domain.

By default, the device is in the VTP no-management-domain state until it receives an advertisement for a domain over a trunk link (a link that carries the traffic of multiple VLANs) or until you configure a domain name. Until the management domain name is specified or learned, you cannot create or modify VLANs on a VTP server, and VLAN information is not propagated over the network.

If the device receives a VTP advertisement over a trunk link, it inherits the management domain name and the VTP configuration revision number. The device then ignores advertisements with a different domain name or an earlier configuration revision number.

When you make a change to the VLAN configuration on a VTP server, the change is propagated to all devices in the VTP domain. VTP advertisements are sent over all IEEE trunk connections, including IEEE 802.1Q. VTP dynamically maps VLANs with unique names and internal index associates across multiple LAN types. Mapping eliminates excessive device administration required from network administrators.

If you configure a device for VTP transparent mode, you can create and modify VLANs, but the changes are not sent to other devices in the domain, and they affect only the individual device. However, configuration changes made when the device is in this mode are saved in the device running configuration and can be saved to the device startup configuration file.

VTP Modes

Table 164: VTP Modes

VTP Mode	Description
VTP server	<p>In VTP server mode, you can create, modify, and delete VLANs, and specify other configuration parameters (such as the VTP version) for the entire VTP domain. VTP servers advertise their VLAN configurations to other devices in the same VTP domain and synchronize their VLAN configurations with other devices based on advertisements received over trunk links.</p> <p>VTP server is the default mode.</p> <p>In VTP server mode, VLAN configurations are saved in NVRAM. If the device detects a failure while writing a configuration to NVRAM, VTP mode automatically changes from server mode to client mode. If this happens, the device cannot be returned to VTP server mode until the NVRAM is functioning.</p>
VTP client	<p>A VTP client functions like a VTP server and transmits and receives VTP updates on its trunks, but you cannot create, change, or delete VLANs on a VTP client. VLANs are configured on another device in the domain that is in server mode.</p> <p>In VTP versions 1 and 2 in VTP client mode, VLAN configurations are not saved in NVRAM. In VTP version 3, VLAN configurations are saved in NVRAM in client mode.</p>

VTP Mode	Description
VTP transparent	<p>VTP transparent devices do not participate in VTP. A VTP transparent device does not advertise its VLAN configuration and does not synchronize its VLAN configuration based on received advertisements. However, in VTP version 2 or version 3, transparent devices do forward VTP advertisements that they receive from other devices through their trunk interfaces. You can create, modify, and delete VLANs on a device in VTP transparent mode.</p> <p>In VTP versions 1 and 2, the device must be in VTP transparent mode when you create private VLANs and when they are configured, you should not change the VTP mode from transparent to client or server mode. VTP version 3 also supports private VLANs in client and server modes. When private VLANs are configured, do not change the VTP mode from transparent to client or server mode.</p> <p>When the device is in VTP transparent mode, the VTP and VLAN configurations are saved in NVRAM, but they are not advertised to other devices. In this mode, VTP mode and domain name are saved in the device running configuration, and you can save this information in the device startup configuration file by using the copy running-config startup-config privileged EXEC command.</p>
VTP off	A device in VTP off mode functions in the same manner as a VTP transparent device, except that it does not forward VTP advertisements on trunks.

VTP Advertisements

Each device in the VTP domain sends periodic global configuration advertisements from each trunk port to a reserved multicast address. Neighboring devices receive these advertisements and update their VTP and VLAN configurations as necessary.

Because trunk ports send and receive VTP advertisements, you must ensure that at least one trunk port is configured on the switch stack and that this trunk port is connected to the trunk port of another switch. Otherwise, the switch cannot receive any VTP advertisements.

VTP advertisements distribute this global domain information:

- VTP domain name
- VTP configuration revision number
- Update identity and update timestamp
- MD5 digest VLAN configuration, including maximum transmission unit (MTU) size for each VLAN

- Frame format

VTP advertisements distribute this VLAN information for each configured VLAN:

- VLAN IDs (including IEEE 802.1Q)
- VLAN name
- VLAN type
- VLAN state
- Additional VLAN configuration information specific to the VLAN type

In VTP version 3, VTP advertisements also include the primary server ID, an instance number, and a start index.

VTP Version 2

If you use VTP in your network, you must decide which version of VTP to use. By default, VTP operates in version 1.

VTP version 2 supports these features that are not supported in version 1:

- Token Ring support—VTP version 2 supports Token Ring Bridge Relay Function (TrBRF) and Token Ring Concentrator Relay Function (TrCRF) VLANs.
- Unrecognized Type-Length-Value (TLV) support—A VTP server or client propagates configuration changes to its other trunks, even for TLVs it is not able to parse. The unrecognized TLV is saved in NVRAM when the device is operating in VTP server mode.
- Version-Dependent Transparent Mode—In VTP version 1, a VTP transparent device inspects VTP messages for the domain name and version and forwards a message only if the version and domain name match. Although VTP version 2 supports only one domain, a VTP version 2 transparent device forwards a message only when the domain name matches.
- Consistency Checks—In VTP version 2, VLAN consistency checks (such as VLAN names and values) are performed only when you enter new information through the CLI or SNMP. Consistency checks are not performed when new information is obtained from a VTP message or when information is read from NVRAM. If the MD5 digest on a received VTP message is correct, its information is accepted.

VTP Version 3

VTP version 3 supports these features that are not supported in version 1 or version 2:

- Enhanced authentication—You can configure the authentication as **hidden** or **secret**. When **hidden**, the secret key from the password string is saved in the VLAN database file, but it does not appear in plain text in the configuration. Instead, the key associated with the password is saved in hexadecimal format in the running configuration. You must reenter the password if you enter a takeover command in the domain. When you enter the **secret** keyword, you can directly configure the password secret key.
- Support for extended range VLAN (VLANs 1006 to 4094) database propagation—VTP versions 1 and 2 propagate only VLANs 1 to 1005.



Note VTP pruning still applies only to VLANs 1 to 1005, and VLANs 1002 to 1005 are still reserved and cannot be modified.

- Private VLAN support.
- Support for any database in a domain—In addition to propagating VTP information, version 3 can propagate Multiple Spanning Tree (MST) protocol database information. A separate instance of the VTP protocol runs for each application that uses VTP.
- VTP primary server and VTP secondary servers—A VTP primary server updates the database information and sends updates that are honored by all devices in the system. A VTP secondary server can only back up the updated VTP configurations received from the primary server to its NVRAM.

By default, all devices come up as secondary servers. You can enter the **vtp primary** privileged EXEC command to specify a primary server. Primary server status is only needed for database updates when the administrator issues a takeover message in the domain. You can have a working VTP domain without any primary servers. Primary server status is lost if the device reloads or domain parameters change, even when a password is configured on the device.

VTP Pruning

VTP pruning increases network available bandwidth by restricting flooded traffic to those trunk links that the traffic must use to reach the destination devices. Without VTP pruning, a device floods broadcast, multicast, and unknown unicast traffic across all trunk links within a VTP domain even though receiving devices might discard them. VTP pruning is disabled by default.

VTP pruning blocks unneeded flooded traffic to VLANs on trunk ports that are included in the pruning-eligible list. Only VLANs included in the pruning-eligible list can be pruned. By default, VLANs 2 through 1001 are pruning eligible device trunk ports. If the VLANs are configured as pruning-ineligible, the flooding continues. VTP pruning is supported in all VTP versions.

With VTP versions 1 and 2, when you enable pruning on the VTP server, it is enabled for the entire VTP domain. In VTP version 3, you must manually enable pruning on each device in the domain. Making VLANs pruning-eligible or pruning-ineligible affects pruning eligibility for those VLANs on that trunk only (not on all devices in the VTP domain).

VTP pruning takes effect several seconds after you enable it. VTP pruning does not prune traffic from VLANs that are pruning-ineligible. VLAN 1 and VLANs 1002 to 1005 are always pruning-ineligible; traffic from these VLANs cannot be pruned. Extended-range VLANs (VLAN IDs higher than 1005) are also pruning-ineligible.

VTP and Switch Stacks

VTP configuration is the same in all members of a switch stack. When the switch stack is in VTP server or client mode, all switches in the stack carry the same VTP configuration. When VTP mode is transparent, the stack is not taking part in VTP.

- When a switch joins the stack, it inherits the VTP and VLAN properties of the active stack.
- All VTP updates are carried across the stack.

- When VTP mode is changed in a switch in the stack, the other switches in the stack also change VTP mode, and the switch VLAN database remains consistent.

VTP version 3 functions the same on a standalone switch or a stack except when the switch stack is the primary server for the VTP database. In this case, the MAC address of the active stack is used as the primary server ID. If the active stack reloads or is powered off, a new active stack is elected.

- If you do not configure the persistent MAC address feature (by entering the **stack-mac persistent timer** [0 | *time-value*] global configuration command, when the new active stack is elected, it sends a takeover message with the new active stack MAC address as the primary server.
- If persistent MAC address is configured, the new active stack waits for the configured **stack-mac persistent timer** value. If the previous active stack does not rejoin the stack during this time, then the new active stack issues the takeover message.

VTP Configuration Guidelines

Configuration Requirements

When you configure VTP, you must configure a trunk port so that the switch can send and receive VTP advertisements to and from other switches in the domain.

If you are configuring VTP on a cluster member switch to a VLAN, use the **command** privileged EXEC command to log in to the member switch. For more information about the command, see the command reference for this release.

In VTP versions 1 and 2, when you configure extended-range VLANs on the switch, the switch must be in VTP transparent mode. VTP version 3 also supports creating extended-range VLANs in client or server mode.

VTP versions 1 and 2 do not support private VLANs. VTP version 3 does support private VLANs. If you configure private VLANs, the switch must be in VTP transparent mode. When private VLANs are configured on the switch, do not change the VTP mode from transparent to client or server mode.

VTP Settings

The VTP information is saved in the VTP VLAN database. When VTP mode is transparent, the VTP domain name and mode are also saved in the device running configuration file, and you can save it in the device startup configuration file by entering the **copy running-config startup-config** privileged EXEC command. You must use this command if you want to save VTP mode as transparent, even if the device resets.

When you save VTP information in the device startup configuration file and reboot the device, the device configuration is selected as follows:

- If the VTP mode is transparent in the startup configuration and the VLAN database and the VTP domain name from the VLAN database matches that in the startup configuration file, the VLAN database is ignored (cleared), and the VTP and VLAN configurations in the startup configuration file are used. The VLAN database revision number remains unchanged in the VLAN database.
- If the VTP mode or domain name in the startup configuration do not match the VLAN database, the domain name and VTP mode and configuration for VLAN IDs 1 to 1005 use the VLAN database information.

Domain Names for Configuring VTP

When configuring VTP for the first time, you must always assign a domain name. You must configure all devices in the VTP domain with the same domain name. Devices in VTP transparent mode do not exchange VTP messages with other devices, and you do not need to configure a VTP domain name for them.



Note If the NVRAM and DRAM storage is sufficient, all devices in a VTP domain should be in VTP server mode.



Caution Do not configure a VTP domain if all devices are operating in VTP client mode. If you configure the domain, it is impossible to make changes to the VLAN configuration of that domain. Make sure that you configure at least one device in the VTP domain for VTP server mode.

Passwords for the VTP Domain

You can configure a password for the VTP domain, but it is not required. If you do configure a domain password, all domain devices must share the same password and you must configure the password on each device in the management domain. Devices without a password or with the wrong password reject VTP advertisements.

If you configure a VTP password for a domain, a device that is booted without a VTP configuration does not accept VTP advertisements until you configure it with the correct password. After the configuration, the device accepts the next VTP advertisement that uses the same password and domain name in the advertisement.

If you are adding a new device to an existing network with VTP capability, the new device learns the domain name only after the applicable password has been configured on it.



Caution When you configure a VTP domain password, the management domain does not function properly if you do not assign a management domain password to each device in the domain.

VTP Version

Follow these guidelines when deciding which VTP version to implement:

- All devices in a VTP domain must have the same domain name, but they do not need to run the same VTP version.
- A VTP version 2-capable device can operate in the same VTP domain as a device running VTP version 1 if version 2 is disabled on the version 2-capable device (version 2 is disabled by default).
- If a device running VTP version 1, but capable of running VTP version 2, receives VTP version 3 advertisements, it automatically moves to VTP version 2.
- If a device running VTP version 3 is connected to a device running VTP version 1, the VTP version 1 device moves to VTP version 2, and the VTP version 3 device sends scaled-down versions of the VTP packets so that the VTP version 2 device can update its database.
- A device running VTP version 3 cannot move to version 1 or 2 if it has extended VLANs.

- Do not enable VTP version 2 on a device unless all of the devices in the same VTP domain are version-2-capable. When you enable version 2 on a device, all of the version-2-capable devices in the domain enable version 2. If there is a version 1-only device, it does not exchange VTP information with devices that have version 2 enabled.
- Cisco recommends placing VTP version 1 and 2 devices at the edge of the network because they do not forward VTP version 3 advertisements.
- If there are TrBRF and TrCRF Token Ring networks in your environment, you must enable VTP version 2 or version 3 for Token Ring VLAN switching to function properly. To run Token Ring and Token Ring-Net, disable VTP version 2.
- VTP version 1 and version 2 do not propagate configuration information for extended range VLANs (VLANs 1006 to 4094). You must configure these VLANs manually on each device. VTP version 3 supports extended-range VLANs and support for extended range VLAN database propagation.
- When a VTP version 3 device trunk port receives messages from a VTP version 2 device, it sends a scaled-down version of the VLAN database on that particular trunk in VTP version 2 format. A VTP version 3 device does not send VTP version 2-formatted packets on a trunk unless it first receives VTP version 2 packets on that trunk port.
- When a VTP version 3 device detects a VTP version 2 device on a trunk port, it continues to send VTP version 3 packets, in addition to VTP version 2 packets, to allow both kinds of neighbors to coexist on the same trunk.
- A VTP version 3 device does not accept configuration information from a VTP version 2 or version 1 device.
- Two VTP version 3 regions can only communicate in transparent mode over a VTP version 1 or version 2 region.
- Devices that are only VTP version 1 capable cannot interoperate with VTP version 3 devices.
- VTP version 1 and version 2 do not propagate configuration information for extended range VLANs (VLANs 1006 to 4094). You must manually configure these VLANs on each device.
- If you configure the device for VTP client mode, the device does not create the VLAN database file (vlan.dat). If the device is then powered off, it resets the VTP configuration to the default. To keep the VTP configuration with VTP client mode after the device restarts, you must first configure the VTP domain name before the VTP mode.



Caution If all devices are operating in VTP client mode, do not configure a VTP domain name. If you do, it is impossible to make changes to the VLAN configuration of that domain. Therefore, make sure you configure at least one device as a VTP server.

Default VTP Configuration

The following table shows the default VTP configuration.

Table 165: Default VTP Configuration

Feature	Default Setting
VTP domain name	Null
VTP mode (VTP version 1 and version 2)	Server
VTP mode (VTP version 3)	The mode is the same as the mode in VTP version 1 or 2 before conversion to version 3.
VTP version	Version 1
MST database mode	Transparent
VTP version 3 server type	Secondary
VTP password	None
VTP pruning	Disabled

How to Configure VTP

Configuring VTP Mode

You can configure VTP mode as one of these:

- VTP server mode—In VTP server mode, you can change the VLAN configuration and have it propagated throughout the network.
- VTP client mode—In VTP client mode, you cannot change its VLAN configuration. The client device receives VTP updates from a VTP server in the VTP domain and then modifies its configuration accordingly.
- VTP transparent mode—In VTP transparent mode, VTP is disabled on the device. The device does not send VTP updates and does not act on VTP updates received from other device. However, a VTP transparent device running VTP version 2 does forward received VTP advertisements on its trunk links.
- VTP off mode—VTP off mode is the same as VTP transparent mode except that VTP advertisements are not forwarded.

When you configure a domain name, it cannot be removed; you can only reassign a device to a different domain.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **vtp domain** *domain-name*
4. **vtp mode** {**client** | **server** | **transparent** | **off**} {**vlan** | **mst** | **unknown**}
5. **vtp password** *password*

6. `end`
7. `show vtp status`
8. `copy running-config startup-config`

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p><code>enable</code></p> <p>Example:</p> <pre>Device> enable</pre>	<p>Enables privileged EXEC mode.</p> <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	<p><code>configure terminal</code></p> <p>Example:</p> <pre>Device# configure terminal</pre>	<p>Enters global configuration mode.</p>
Step 3	<p><code>vtp domain domain-name</code></p> <p>Example:</p> <pre>Device(config)# vtp domain eng_group</pre>	<p>Configures the VTP administrative-domain name. The name can be 1 to 32 characters. All devices operating in VTP server or client mode under the same administrative responsibility must be configured with the same domain name.</p> <p>This command is optional for modes other than server mode. VTP server mode requires a domain name. If the device has a trunk connection to a VTP domain, the device learns the domain name from the VTP server in the domain.</p> <p>You should configure the VTP domain before configuring other VTP parameters.</p>
Step 4	<p><code>vtp mode {client server transparent off} {vlan mst unknown}</code></p> <p>Example:</p> <pre>Device(config)# vtp mode server</pre>	<p>Configures the device for VTP mode (client, server, transparent, or off).</p> <ul style="list-style-type: none"> • vlan—The VLAN database is the default if none are configured. • mst—The multiple spanning tree (MST) database. • unknown—An unknown database type. <p>Note To return a switch in another mode to VTP server mode, use the no vtp mode global configuration command.</p>
Step 5	<p><code>vtp password password</code></p> <p>Example:</p> <pre>Device(config)# vtp password mypassword</pre>	<p>(Optional) Sets the password for the VTP domain. The password can be 8 to 64 characters. If you configure a VTP password, the VTP domain does not function properly if you do not assign the same password to each device in the domain.</p>

	Command or Action	Purpose
		Note To return the switch to a no-password state, use the no vtp password global configuration command.
Step 6	end Example: Device(config) # end	Returns to privileged EXEC mode.
Step 7	show vtp status Example: Device# show vtp status	Verifies your entries in the <i>VTP Operating Mode</i> and the <i>VTP Domain Name</i> fields of the display.
Step 8	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves the configuration in the startup configuration file. Only VTP mode and domain name are saved in the device running configuration and can be copied to the startup configuration file.

Configuring a VTP Version 3 Password

You can configure a VTP version 3 password on the device.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **vtp version 3**
4. **vtp password *password* [hidden | secret]**
5. **end**
6. **show vtp password**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. • Enter your password if prompted.

	Command or Action	Purpose
Step 2	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode.
Step 3	vtp version 3 Example: Device(config)# <code>vtp version 3</code>	Enables VTP version 3 on the device. The default is VTP version 1.
Step 4	vtp password <i>password</i> [hidden secret] Example: Device(config)# <code>vtp password mypassword hidden</code>	(Optional) Sets the password for the VTP domain. The password can be 8 to 64 characters. <ul style="list-style-type: none"> • (Optional) hidden—Saves the secret key generated from the password string in the nvram:vlan.dat file. If you configure a takeover by configuring a VTP primary server, you are prompted to reenter the password. • (Optional) secret—Directly configures the password. The secret password must contain 32 hexadecimal characters. <p>Note To clear the password, enter the no vtp password global configuration command.</p>
Step 5	end Example: Device(config)# <code>end</code>	Returns to privileged EXEC mode.
Step 6	show vtp password Example: Device# <code>show vtp password</code>	Verifies your entries. The output appears like this: VTP password: 89914640C8D90868B6A0D8103847A733
Step 7	copy running-config startup-config Example: Device# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Configuring a VTP Version 3 Primary Server

When you configure a VTP server as a VTP primary server, the takeover operation starts.

SUMMARY STEPS

1. `vtp version 3`
2. `vtp primary [vlan | mst] [force]`

DETAILED STEPS

	Command or Action	Purpose
Step 1	vtp version 3 Example: Device(config)# <code>vtp version 3</code>	Enables VTP version 3 on the device. The default is VTP version 1.
Step 2	vtp primary [vlan mst] [force] Example: Device# <code>vtp primary vlan force</code>	Changes the operational state of a device from a secondary server (the default) to a primary server and advertises the configuration to the domain. If the device password is configured as hidden , you are prompted to reenter the password. <ul style="list-style-type: none"> • (Optional) vlan—Selects the VLAN database as the takeover feature. This is the default. • (Optional) mst—Selects the multiple spanning tree (MST) database as the takeover feature. • (Optional) force—Overwrites the configuration of any conflicting servers. If you do not enter force, you are prompted for confirmation before the takeover.

Enabling the VTP Version

VTP version 2 and version 3 are disabled by default.

- When you enable VTP version 2 on a device, every VTP version 2-capable device in the VTP domain enables version 2. To enable VTP version 3, you must manually configure it on each device.
- With VTP versions 1 and 2, you can configure the version only on devices in VTP server or transparent mode. If a device is running VTP version 3, you can change to version 2 when the device is in client mode if no extended VLANs exist, no private VLANs exist, and no hidden password was configured.



Caution VTP version 1 and VTP version 2 are not interoperable on devices in the same VTP domain. Do not enable VTP version 2 unless every device in the VTP domain supports version 2.

- In TrCRF and TrBRF Token Ring environments, you must enable VTP version 2 or VTP version 3 for Token Ring VLAN switching to function properly. For Token Ring and Token Ring-Net media, disable VTP version 2.



Caution In VTP version 3, both the primary and secondary servers can exist on an instance in the domain.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **vtp version {1 | 2 | 3}**
4. **end**
5. **show vtp status**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	vtp version {1 2 3} Example: Device(config)# vtp version 2	Enables the VTP version on the device. The default is VTP version 1. Note To return to the default VTP version 1, use the no vtp version global configuration command.
Step 4	end Example: Device(config)# end	Returns to privileged EXEC mode.
Step 5	show vtp status Example:	Verifies that the configured VTP version is enabled.

	Command or Action	Purpose
	Device# <code>show vtp status</code>	
Step 6	copy running-config startup-config Example: Device# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Enabling VTP Pruning

Before you begin

VTP pruning is not designed to function in VTP transparent mode. If one or more devices in the network are in VTP transparent mode, you should do one of these actions:

- Turn off VTP pruning in the entire network.
- Turn off VTP pruning by making all VLANs on the trunk of the device upstream to the VTP transparent device pruning ineligible.

To configure VTP pruning on an interface, use the **switchport trunk pruning vlan** interface configuration command. VTP pruning operates when an interface is trunking. You can set VLAN pruning-eligibility, whether or not VTP pruning is enabled for the VTP domain, whether or not any given VLAN exists, and whether or not the interface is currently trunking.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **vtp pruning**
4. **end**
5. **show vtp status**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> <code>enable</code>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example:	Enters global configuration mode.

	Command or Action	Purpose
	Device# <code>configure terminal</code>	
Step 3	vtp pruning Example: Device(config)# <code>vtp pruning</code>	Enables pruning in the VTP administrative domain. By default, pruning is disabled. You need to enable pruning on only one device in VTP server mode. Note To disable VTP pruning, use the no vtp pruning global configuration command.
Step 4	end Example: Device(config)# <code>end</code>	Returns to privileged EXEC mode.
Step 5	show vtp status Example: Device# <code>show vtp status</code>	Verifies your entries in the <i>VTP Pruning Mode</i> field of the display.

Configuring VTP on a Per-Port Basis

With VTP version 3, you can enable or disable VTP on a per-port basis. You can enable VTP only on ports that are in trunk mode. Incoming and outgoing VTP traffic are blocked, not forwarded.

SUMMARY STEPS

1. `enable`
2. `configure terminal`
3. `interface interface-id`
4. `vtp`
5. `end`
6. `show running-config interface interface-id`
7. `show vtp status`

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> <code>enable</code>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.

	Command or Action	Purpose
Step 2	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode.
Step 3	interface <i>interface-id</i> Example: Device(config)# <code>interface gigabitethernet0/1</code>	Identifies an interface, and enters interface configuration mode.
Step 4	vtp Example: Device(config-if)# <code>vtp</code>	Enables VTP on the specified port. Note To disable VTP on the interface, use the no vtp interface configuration command.
Step 5	end Example: Device(config)# <code>end</code>	Returns to privileged EXEC mode.
Step 6	show running-config interface <i>interface-id</i> Example: Device# <code>show running-config interface gigabitethernet 1/0/1</code>	Verifies the change to the port.
Step 7	show vtp status Example: Device# <code>show vtp status</code>	Verifies the configuration.

Adding a VTP Client to a VTP Domain

Follow these steps to verify and reset the VTP configuration revision number on a device *before* adding it to a VTP domain.

Before you begin

Before adding a VTP client to a VTP domain, always verify that its VTP configuration revision number is *lower* than the configuration revision number of the other devices in the VTP domain. Devices in a VTP domain always use the VLAN configuration of the device with the highest VTP configuration revision number. With VTP versions 1 and 2, adding a device that has a revision number higher than the revision number in

the VTP domain can erase all VLAN information from the VTP server and VTP domain. With VTP version 3, the VLAN information is not erased.

You can use the **vtp mode transparent** global configuration command to disable VTP on the device and then to change its VLAN information without affecting the other devices in the VTP domain.

SUMMARY STEPS

1. **enable**
2. **show vtp status**
3. **configure terminal**
4. **vtp domain *domain-name***
5. **end**
6. **show vtp status**
7. **configure terminal**
8. **vtp domain *domain-name***
9. **end**
10. **show vtp status**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	show vtp status Example: Device# show vtp status	Checks the VTP configuration revision number. If the number is 0, add the device to the VTP domain. If the number is greater than 0, follow these substeps: <ul style="list-style-type: none"> • Write down the domain name. • Write down the configuration revision number. • Continue with the next steps to reset the device configuration revision number.
Step 3	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 4	vtp domain <i>domain-name</i> Example: Device(config)# vtp domain domain123	Changes the domain name from the original one displayed in Step 1 to a new name.

	Command or Action	Purpose
Step 5	end Example: Device(config)# end	Returns to privileged EXEC mode. The VLAN information on the device is updated and the configuration revision number is reset to 0.
Step 6	show vtp status Example: Device# show vtp status	Verifies that the configuration revision number has been reset to 0.
Step 7	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 8	vtp domain <i>domain-name</i> Example: Device(config)# vtp domain domain012	Enters the original domain name on the device
Step 9	end Example: Device(config)# end	Returns to privileged EXEC mode. The VLAN information on the device is updated.
Step 10	show vtp status Example: Device# show vtp status	(Optional) Verifies that the domain name is the same as in Step 1 and that the configuration revision number is 0.

Monitoring VTP

This section describes commands used to display and monitor the VTP configuration.

You monitor VTP by displaying VTP configuration information: the domain name, the current VTP revision, and the number of VLANs. You can also display statistics about the advertisements sent and received by the device.

Table 166: VTP Monitoring Commands

Command	Purpose
show vtp counters	Displays counters about VTP messages that have been sent and received.
show vtp devices [conflict]	Displays information about all VTP version 3 devices in the domain. Conflicts are VTP version 3 devices with conflicting primary servers. The show vtp devices command does not display information when the device is in transparent or off mode.
show vtp interface [interface-id]	Displays VTP status and configuration for all interfaces or the specified interface.
show vtp password	Displays the VTP password. The form of the password displayed depends on whether or not the hidden keyword was entered and if encryption is enabled on the device.
show vtp status	Displays the VTP device configuration information.

Configuration Examples for VTP

Example: Configuring the Switch as a VTP Server

This example shows how to configure the switch as a VTP server with the domain name *eng_group* and the password *mypassword*:

```
Switch(config)# vtp domain eng_group
Setting VTP domain name to eng_group.

Switch(config)# vtp mode server

Setting device to VTP Server mode for VLANs.
Switch(config)# vtp password mypassword

Setting device VLAN database password to mypassword.

Switch(config)# end
```

Example: Configuring a Hidden Password

This example shows how to configure a hidden password and how it appears.

```
Switch(config)# vtp password mypassword hidden
Generating the secret associated to the password.
Switch(config)# end
Switch# show vtp password
VTP password: 89914640C8D90868B6A0D8103847A733
```

Example: Configuring a VTP Version 3 Primary Server

This example shows how to configure a switch as the primary server for the VLAN database (the default) when a hidden or secret password was configured:

```
Switch# vtp primary vlan

Enter VTP password: mypassword
This switch is becoming Primary server for vlan feature in the VTP domain
VTP Database Conf Switch ID Primary Server Revision System Name
-----
VLANDB Yes 00d0.00b8.1400=00d0.00b8.1400 1 stp7
Do you want to continue (y/n) [n]? y
```

Example: Configuring VTP on a Per-Port Basis

This example shows how to configure VTP on a per-port basis:

```
Switch(config)# interface gigabitethernet 1/0/1
Switch(config-if)# vtp
Switch(config-if)# end
```

Where to Go Next

After configuring VTP, you can configure the following:

- VLANs
- VLAN trunking
- Private VLANs
- VLAN Membership Policy Server (VMPS)
- Tunneling
- Voice VLANs

Additional References

Related Documents

Related Topic	Document Title
For complete syntax and usage information for the commands used in this chapter.	<i>Catalyst 2960-XR Switch VLAN Management Command Reference</i>

Standards and RFCs

Standard/RFC	Title
—	—

MIBs

MIB	MIBs Link
All the supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
<p>The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies.</p> <p>To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds.</p> <p>Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.</p>	http://www.cisco.com/support

Feature History and Information for VTP

Release	Modification
Cisco IOS Release 15.0(2)EX1	This feature was introduced.



CHAPTER 85

VLANs

- [Prerequisites for VLANs, on page 1699](#)
- [Restrictions for VLANs, on page 1699](#)
- [Information About VLANs, on page 1699](#)
- [How to Configure VLANs, on page 1706](#)
- [Monitoring VLANs, on page 1715](#)
- [Configuration Examples, on page 1716](#)
- [Where to Go Next, on page 1716](#)
- [Additional References, on page 1717](#)
- [Feature History and Information for VLAN, on page 1717](#)

Prerequisites for VLANs

The following are prerequisites and considerations for configuring VLANs:

- The switch supports 1005 VLANs when running the IP Lite image.
- The switch supports up to 1000 VLANs running the LAN Base image.
- The switch supports 256 SVIs when running the IP Lite image.

Restrictions for VLANs

The following are the restrictions for configuring VLANs:

- The switch supports homogeneous stacking, but does not support mixed stacking.

Information About VLANs

Logical Networks

A VLAN is a switched network that is logically segmented by function, project team, or application, without regard to the physical locations of the users. VLANs have the same attributes as physical LANs, but you can

group end stations even if they are not physically located on the same LAN segment. Any device port can belong to a VLAN, and unicast, broadcast, and multicast packets are forwarded and flooded only to end stations in the VLAN. Each VLAN is considered a logical network, and packets destined for stations that do not belong to the VLAN must be forwarded through a router or a device supporting fallback bridging. Because a VLAN is considered a separate logical network, it contains its own bridge Management Information Base (MIB) information and can support its own implementation of spanning tree.

VLANs are often associated with IP subnetworks. For example, all the end stations in a particular IP subnet belong to the same VLAN. Interface VLAN membership on the device is assigned manually on an interface-by-interface basis. When you assign device interfaces to VLANs by using this method, it is known as interface-based, or static, VLAN membership.

Traffic between VLANs must be routed or fallback bridged.

The device can route traffic between VLANs by using device virtual interfaces (SVIs). An SVI must be explicitly configured and assigned an IP address to route traffic between VLANs.

Supported VLANs

The switch supports VLANs in VTP client, server, and transparent modes. VLANs are identified by a number from 1 to 4094. VLAN IDs 1002 through 1005 are reserved for Token Ring and FDDI VLANs. VTP version 1 and version 2 support only normal-range VLANs (VLAN IDs 1 to 1005). In these versions, the switch must be in VTP transparent mode when you create VLAN IDs from 1006 to 4094. VTP version 3 supports the entire VLAN range (VLANs 1 to 4094). Extended range VLANs (VLANs 1006 to 4094) are supported only in VTP version 3. You cannot convert from VTP version 3 to VTP version 2 if extended VLANs are configured in the domain.

The switch or switch stack supports a total of 1005 (normal range and extended range) VLANs. However, the number of routed ports, SVIs, and other configured features affects the use of the switch hardware.

The switch or switch stack supports a total of 255 VLANs when running the LAN base feature set. However, the number of routed ports, SVIs, and other configured features affects the use of the switch hardware.

The switch supports per-VLAN spanning-tree plus (PVST+) or rapid PVST+ with a maximum of 128 spanning-tree instances. One spanning-tree instance is allowed per VLAN.

The switch supports IEEE 802.1Q trunking methods for sending VLAN traffic over Ethernet ports.

VLAN Port Membership Modes

You configure a port to belong to a VLAN by assigning a membership mode that specifies the kind of traffic the port carries and the number of VLANs to which it can belong.

When a port belongs to a VLAN, the device learns and manages the addresses associated with the port on a per-VLAN basis.

Table 167: Port Membership Modes and Characteristics

Membership Mode	VLAN Membership Characteristics	VTP Characteristics
Static-access	A static-access port can belong to one VLAN and is manually assigned to that VLAN.	VTP is not required. If you do not want VTP to globally propagate information, set the VTP mode to transparent. To participate in VTP, there must be at least one trunk port on the device connected to a trunk port of a second device.
Trunk (IEEE 802.1Q) : <ul style="list-style-type: none"> IEEE 802.1Q—Industry-standard trunking encapsulation. 	A trunk port is a member of all VLANs by default, including extended-range VLANs, but membership can be limited by configuring the allowed-VLAN list. You can also modify the pruning-eligible list to block flooded traffic to VLANs on trunk ports that are included in the list.	VTP is recommended but not required. VTP maintains VLAN configuration consistency by managing the addition, deletion, and renaming of VLANs on a network-wide basis. VTP exchanges VLAN configuration messages with other devices over trunk links.
Dynamic access	A dynamic-access port can belong to one VLAN (VLAN ID 1 to 4094) and is dynamically assigned by a VLAN Member Policy Server (VMPS). You can have dynamic-access ports and trunk ports on the same device, but you must connect the dynamic-access port to an end station or hub and not to another device.	VTP is required. Configure the VMPS and the client with the same VTP domain name. To participate in VTP, at least one trunk port on the device must be connected to a trunk port of a second device .
Voice VLAN	A voice VLAN port is an access port attached to a Cisco IP Phone, configured to use one VLAN for voice traffic and another VLAN for data traffic from a device attached to the phone.	VTP is not required; it has no effect on a voice VLAN.

VLAN Configuration Files

Configurations for VLAN IDs 1 to 1005 are written to the `vlan.dat` file (VLAN database), and you can display them by entering the **show vlan** privileged EXEC command. The `vlan.dat` file is stored in flash memory. If the VTP mode is transparent, they are also saved in the device running configuration file.

You use the interface configuration mode to define the port membership mode and to add and remove ports from VLANs. The results of these commands are written to the running-configuration file, and you can display the file by entering the **show running-config** privileged EXEC command.

When you save VLAN and VTP information (including extended-range VLAN configuration information) in the startup configuration file and reboot the device, the device configuration is selected as follows:

- If the VTP mode is transparent in the startup configuration, and the VLAN database and the VTP domain name from the VLAN database matches that in the startup configuration file, the VLAN database is

ignored (cleared), and the VTP and VLAN configurations in the startup configuration file are used. The VLAN database revision number remains unchanged in the VLAN database.

- If the VTP mode or domain name in the startup configuration does not match the VLAN database, the domain name and VTP mode and configuration for the VLAN IDs 1 to 1005 use the VLAN database information.
- In VTP versions 1 and 2, if VTP mode is server, the domain name and VLAN configuration for VLAN IDs 1 to 1005 use the VLAN database information. VTP version 3 also supports VLANs 1006 to 4094.
- From image 15.0(02)SE6, on vtp transparent and off modes, vlans get created from startup-config even if they are not applied to the interface.



Note Ensure that you delete the `vlan.dat` file along with the configuration files before you reset the switch configuration using **write erase** command. This ensures that the switch reboots correctly on a reset.

Normal-Range VLAN Overview

Normal-range VLANs are VLANs with VLAN IDs 1 to 1005. If the switch is in VTP server or VTP transparent mode, you can add, modify or remove configurations for VLANs 2 to 1001 in the VLAN database. (VLAN IDs 1 and 1002 to 1005 are automatically created and cannot be removed.)

In VTP versions 1 and 2, the switch must be in VTP transparent mode when you create extended-range VLANs (VLANs with IDs from 1006 to 4094), but these VLANs are not saved in the VLAN database. VTP version 3 supports extended-range VLANs in VTP server and transparent mode.

Configurations for VLAN IDs 1 to 1005 are written to the file `vlan.dat` (VLAN database), and you can display them by entering the **show vlan** privileged EXEC command. The `vlan.dat` file is stored in flash memory. On a switch, the `vlan.dat` file is stored in flash memory on the active stack. Stack members have a `vlan.dat` file that is consistent with the active stack.

Token Ring VLANs

Although the switch does not support Token Ring connections, a remote device such as a Catalyst 5000 series switch with Token Ring connections could be managed from one of the supported switches. Switches running VTP Version 2 advertise information about these Token Ring VLANs:

- Token Ring TrBRF VLANs
- Token Ring TrCRF VLANs



Note For more information on configuring Token Ring VLANs, see the *Catalyst 5000 Series Software Configuration Guide*.

Normal-Range VLANs Configuration Process

You configure VLANs in the **vlan** global configuration command by entering a VLAN ID. Enter a new VLAN ID to create a VLAN, or enter an existing VLAN ID to modify that VLAN. You can use the default

VLAN configuration or enter multiple commands to configure the VLAN. For more information about commands available in this mode, see the **vlan** global configuration command description in the command reference for this release. When you have finished the configuration, you must exit VLAN configuration mode for the configuration to take effect. To display the VLAN configuration, enter the **show vlan** privileged EXEC command.

VLAN Configuration Saving Process

The configurations of VLAN IDs 1 to 1005 are always saved in the VLAN database (vlan.dat file). If the VTP mode is transparent, they are also saved in the switch running configuration file. You can enter the **copy running-config startup-config** privileged EXEC command to save the configuration in the startup configuration file. In a switch stack, the whole stack uses the same vlan.dat file and running configuration. To display the VLAN configuration, enter the **show vlan** privileged EXEC command.

When you save VLAN and VTP information (including extended-range VLAN configuration information) in the startup configuration file and reboot the switch, the switch configuration is selected as follows:

- If the VTP mode is transparent in the startup configuration, and the VLAN database and the VTP domain name from the VLAN database matches that in the startup configuration file, the VLAN database is ignored (cleared), and the VTP and VLAN configurations in the startup configuration file are used. The VLAN database revision number remains unchanged in the VLAN database.
- If the VTP mode or domain name in the startup configuration does not match the VLAN database, the domain name and VTP mode and configuration for the VLAN IDs 1 to 1005 use the VLAN database information.
- In VTP versions 1 and 2, if VTP mode is server, the domain name and VLAN configuration for VLAN IDs 1 to 1005 use the VLAN database information. VTP version 3 also supports VLANs 1006 to 4094.

Normal-Range VLAN Configuration Guidelines

Normal-range VLANs are VLANs with IDs from 1 to 1005.

VTP 1 and 2 only support normal-range VLANs.

Follow these guidelines when creating and modifying normal-range VLANs in your network:

- Normal-range VLANs are identified with a number between 1 and 1001. VLAN numbers 1002 through 1005 are reserved for Token Ring and FDDI VLANs.
- VLAN configurations for VLANs 1 to 1005 are always saved in the VLAN database. If the VTP mode is transparent, VTP and VLAN configurations are also saved in the device running configuration file.
- If the device is in VTP server or VTP transparent mode, you can add, modify or remove configurations for VLANs 2 to 1001 in the VLAN database. (VLAN IDs 1 and 1002 to 1005 are automatically created and cannot be removed.)
- With VTP versions 1 and 2, the device supports VLAN IDs 1006 through 4094 only in VTP transparent mode (VTP disabled). These are extended-range VLANs and configuration options are limited. Extended-range VLANs created in VTP transparent mode are not saved in the VLAN database and are not propagated. VTP version 3 supports extended range VLAN (VLANs 1006 to 4094) database propagation in VTP server mode. If extended VLANs are configured, you cannot convert from VTP version 3 to version 1 or 2.

- Before you can create a VLAN, the device must be in VTP server mode or VTP transparent mode. If the device is a VTP server, you must define a VTP domain or VTP will not function.
- The device does not support Token Ring or FDDI media. The device does not forward FDDI, FDDI-Net, TrCRF, or TrBRF traffic, but it does propagate the VLAN configuration through VTP.
- The device supports 128 spanning tree instances. If a device has more active VLANs than supported spanning-tree instances, spanning tree can be enabled on 128 VLANs and is disabled on the remaining VLANs.

If you have already used all available spanning-tree instances on a device, adding another VLAN anywhere in the VTP domain creates a VLAN on that device that is not running spanning-tree. If you have the default allowed list on the trunk ports of that device (which is to allow all VLANs), the new VLAN is carried on all trunk ports. Depending on the topology of the network, this could create a loop in the new VLAN that would not be broken, particularly if there are several adjacent devices that all have run out of spanning-tree instances. You can prevent this possibility by setting allowed lists on the trunk ports of devices that have used up their allocation of spanning-tree instances.

If the number of VLANs on the device exceeds the number of supported spanning-tree instances, we recommend that you configure the IEEE 802.1s Multiple STP (MSTP) on your device to map multiple VLANs to a single spanning-tree instance.

Extended-Range VLAN Configuration Guidelines

Extended-range VLANs are VLANs with IDs from 1006 to 4094.

VTP 3 only supports extended-range VLANs.

Follow these guidelines when creating extended-range VLANs:

- VLAN IDs in the extended range are not saved in the VLAN database and are not recognized by VTP unless the device is running VTP version 3.
- You cannot include extended-range VLANs in the pruning eligible range.
- For VTP version 1 or 2, you can set the VTP mode to transparent in global configuration mode. You should save this configuration to the startup configuration so that the device boots up in VTP transparent mode. Otherwise, you lose the extended-range VLAN configuration if the device resets. If you create extended-range VLANs in VTP version 3, you cannot convert to VTP version 1 or 2.
- Each routed port on the device creates an internal VLAN for its use. These internal VLANs use extended-range VLAN numbers, and the internal VLAN ID cannot be used for an extended-range VLAN. If you try to create an extended-range VLAN with a VLAN ID that is already allocated as an internal VLAN, an error message is generated, and the command is rejected.



Note Devices running the LAN Base feature set support only static routing on SVIs.

- Because internal VLAN IDs are in the lower part of the extended range, we recommend that you create extended-range VLANs beginning from the highest number (4094) and moving to the lowest (1006) to reduce the possibility of using an internal VLAN ID.
- Before configuring extended-range VLANs, enter the **show vlan internal usage** privileged EXEC command to see which VLANs have been allocated as internal VLANs.

- If necessary, you can shut down the routed port assigned to the internal VLAN, which frees up the internal VLAN, and then create the extended-range VLAN and re-enable the port, which then uses another VLAN as its internal VLAN.
- Although the device or device stack supports a total of 1005 (normal-range and extended-range) VLANs, the number of routed ports, SVIs, and other configured features affects the use of the device hardware. If you try to create an extended-range VLAN and there are not enough hardware resources available, an error message is generated, and the extended-range VLAN is rejected.

Default Ethernet VLAN Configuration

The following table displays the default configuration for Ethernet VLANs.



Note The switch supports Ethernet interfaces exclusively. Because FDDI and Token Ring VLANs are not locally supported, you only configure FDDI and Token Ring media-specific characteristics for VTP global advertisements to other switches.

Table 168: Ethernet VLAN Defaults and Range

Parameter	Default	Range
VLAN ID	1	1 to 4094. Note Extended-range VLANs (VLAN IDs 1006 to 4094) are only saved in the VLAN database in VTP version 3.
VLAN name	VLANxxxx, where xxxx represents four numeric digits (including leading zeros) equal to the VLAN ID number	No range
IEEE 802.10 SAID	100001 (100000 plus the VLAN ID)	1 to 4294967294
IEEE 802.10 SAID	1500	576-18190
Private VLANs	none configured	2 to 1001, 1006 to 4094

Default VLAN Configuration

You can change only the MTU size, private VLAN, and the remote SPAN configuration state on extended-range VLANs; all other characteristics must remain at the default state.

How to Configure VLANs

How to Configure Normal-Range VLANs

You can set these parameters when you create a new normal-range VLAN or modify an existing VLAN in the VLAN database:

- VLAN ID
- VLAN name
- VLAN type
 - Ethernet
 - Fiber Distributed Data Interface [FDDI]
 - FDDI network entity title [NET]
 - TrBRF or TrCRF
 - Token Ring
 - Token Ring-Net
- VLAN state (active or suspended)
- Security Association Identifier (SAID)
- Bridge identification number for TrBRF VLANs
- Ring number for FDDI and TrCRF VLANs
- Parent VLAN number for TrCRF VLANs
- Spanning Tree Protocol (STP) type for TrCRF VLANs
- VLAN number to use when translating from one VLAN type to another

You can cause inconsistency in the VLAN database if you attempt to manually delete the `vlan.dat` file. If you want to modify the VLAN configuration, follow the procedures in this section.

Creating or Modifying an Ethernet VLAN

Each Ethernet VLAN in the VLAN database has a unique, 4-digit ID that can be a number from 1 to 1001. VLAN IDs 1002 to 1005 are reserved for Token Ring and FDDI VLANs. To create a normal-range VLAN to be added to the VLAN database, assign a number and name to the VLAN.



Note With VTP version 1 and 2, if the device is in VTP transparent mode, you can assign VLAN IDs greater than 1006, but they are not added to the VLAN database.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **vlan *vlan-id***
4. **name *vlan-name***
5. **mtu *mtu-size***
6. **remote-span**
7. **end**
8. **show vlan {name *vlan-name* | id *vlan-id*}**
9. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	vlan <i>vlan-id</i> Example: Device(config)# vlan 20	Enters a VLAN ID, and enters VLAN configuration mode. Enter a new VLAN ID to create a VLAN, or enter an existing VLAN ID to modify that VLAN. Note The available VLAN ID range for this command is 1 to 4094.
Step 4	name <i>vlan-name</i> Example: Device(config-vlan)# name test20	(Optional) Enters a name for the VLAN. If no name is entered for the VLAN, the default is to append the <i>vlan-id</i> value with leading zeros to the word VLAN. For example, VLAN0004 is a default VLAN name for VLAN 4.
Step 5	mtu <i>mtu-size</i> Example: Device(config-vlan)# mtu 256	(Optional) Changes the MTU size (or other VLAN characteristic).
Step 6	remote-span Example:	(Optional) Configures the VLAN as the RSPAN VLAN for a remote SPAN session.

	Command or Action	Purpose
	Device(config-vlan) # remote-span	Note To return the VLAN name to the default settings, use the no name , no mtu , or no remote-span commands.
Step 7	end Example: Device(config) # end	Returns to privileged EXEC mode.
Step 8	show vlan {name vlan-name id vlan-id} Example: Device# show vlan name test20 id 20	Verifies your entries.
Step 9	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Deleting a VLAN

When you delete a VLAN from a device that is in VTP server mode, the VLAN is removed from the VLAN database for all devices in the VTP domain. When you delete a VLAN from a device that is in VTP transparent mode, the VLAN is deleted only on that specific device .

You cannot delete the default VLANs for the different media types: Ethernet VLAN 1 and FDDI or Token Ring VLANs 1002 to 1005.



Caution When you delete a VLAN, any ports assigned to that VLAN become inactive. They remain associated with the VLAN (and thus inactive) until you assign them to a new VLAN.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **no vlan vlan-id**
4. **end**
5. **show vlan brief**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	no vlan <i>vlan-id</i> Example: Device(config)# no vlan 4	Removes the VLAN by entering the VLAN ID.
Step 4	end Example: Device(config)# end	Returns to privileged EXEC mode.
Step 5	show vlan brief Example: Device# show vlan brief	Verifies the VLAN removal.
Step 6	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Assigning Static-Access Ports to a VLAN

You can assign a static-access port to a VLAN without having VTP globally propagate VLAN configuration information by disabling VTP (VTP transparent mode).

If you are assigning a port on a cluster member switch to a VLAN, first use the **rcommand** privileged EXEC command to log in to the cluster member switch.

If you assign an interface to a VLAN that does not exist, the new VLAN is created.

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **switchport mode access**
4. **switchport access vlan** *vlan-id*
5. **end**
6. **show running-config interface** *interface-id*
7. **show interfaces** *interface-id* **switchport**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 2	interface <i>interface-id</i> Example: Device(config)# interface gigabitethernet2/0/1	Enters the interface to be added to the VLAN.
Step 3	switchport mode access Example: Device(config-if)# switchport mode access	Defines the VLAN membership mode for the port (Layer 2 access port).
Step 4	switchport access vlan <i>vlan-id</i> Example: Device(config-if)# switchport access vlan 2	Assigns the port to a VLAN. Valid VLAN IDs are 1 to 4094. Note To return an interface to its default configuration, use the default interface interface-id interface configuration command.
Step 5	end Example: Device(config)# end	Returns to privileged EXEC mode.
Step 6	show running-config interface <i>interface-id</i> Example: Device# show running-config interface gigabitethernet2/0/1	Verifies the VLAN membership mode of the interface.

	Command or Action	Purpose
Step 7	<p>show interfaces <i>interface-id</i> switchport</p> <p>Example:</p> <pre>Device# show interfaces gigabitethernet2/0/1 switchport</pre>	Verifies your entries in the <i>Administrative Mode</i> and the <i>Access Mode VLAN</i> fields of the display.

How to Configure Extended-Range VLANs

Extended-range VLANs enable service providers to extend their infrastructure to a greater number of customers. The extended-range VLAN IDs are allowed for any **switchport** commands that allow VLAN IDs.

With VTP version 1 or 2, extended-range VLAN configurations are not stored in the VLAN database, but because VTP mode is transparent, they are stored in the device running configuration file, and you can save the configuration in the startup configuration file. Extended-range VLANs created in VTP version 3 are stored in the VLAN database.

You can change only the MTU size, private VLAN, and the remote SPAN configuration state on extended-range VLANs; all other characteristics must remain at the default state.

Creating an Extended-Range VLAN

You create an extended-range VLAN in global configuration mode by entering the **vlan** global configuration command with a VLAN ID from 1006 to 4094. The extended-range VLAN has the default Ethernet VLAN characteristics and the MTU size, and RSPAN configuration are the only parameters you can change. See the description of the **vlan** global configuration command in the command reference for the default settings of all parameters. In VTP version 1 or 2, if you enter an extended-range VLAN ID when the switch is not in VTP transparent mode, an error message is generated when you exit VLAN configuration mode, and the extended-range VLAN is not created.

In VTP version 1 and 2, extended-range VLANs are not saved in the VLAN database; they are saved in the switch running configuration file. You can save the extended-range VLAN configuration in the switch startup configuration file by using the **copy running-config startup-config** privileged EXEC command. VTP version 3 saves extended-range VLANs in the VLAN database.



Note Before you create an extended-range VLAN, you can verify that the VLAN ID is not used internally by entering the **show vlan internal usage** privileged EXEC command. If the VLAN ID is used internally and you want to free it up, go to the *Creating an Extended-Range VLAN with an Internal VLAN ID* before creating the extended-range VLAN.

SUMMARY STEPS

1. **configure terminal**
2. **vtp mode transparent**
3. **vlan** *vlan-id*
4. **mtu** *mtu size*
5. **remote-span**

6. **end**
7. **show vlan id *vlan-id***
8. **copy running-config startup config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 2	vtp mode transparent Example: Device(config)# vtp mode transparent	Configures the device for VTP transparent mode, disabling VTP. Note This step is not required for VTP version 3.
Step 3	vlan <i>vlan-id</i> Example: Device(config)# vlan 2000 Device(config-vlan)#	Enters an extended-range VLAN ID and enters VLAN configuration mode. The range is 1006 to 4094. Note To delete an extended-range VLAN, use the no vlan <i>vlan-id</i> global configuration command.
Step 4	mtu <i>mtu size</i> Example: Device(config-vlan)# mtu 1024	Modifies the VLAN by changing the MTU size.
Step 5	remote-span Example: Device(config-vlan)# remote-span	(Optional) Configures the VLAN as the RSPAN VLAN.
Step 6	end Example: Device(config)# end	Returns to privileged EXEC mode.
Step 7	show vlan id <i>vlan-id</i> Example: Device# show vlan id 2000	Verifies that the VLAN has been created.
Step 8	copy running-config startup config	Saves your entries in the device startup configuration file.

	Command or Action	Purpose
	<p>Example:</p> <pre>Device# copy running-config startup-config</pre>	<p>To save an extended-range VLAN configuration, you need to save the VTP transparent mode configuration and the extended-range VLAN configuration in the device startup configuration file. Otherwise, if the device resets, it will default to VTP server mode, and the extended-range VLAN IDs will not be saved.</p> <p>Note This step is not required for VTP version 3 because VLANs are saved in the VLAN database.</p> <p>The procedure for assigning static-access ports to an extended-range VLAN is the same as for normal-range VLANs.</p>

Creating an Extended-Range VLAN with an Internal VLAN ID

If you enter an extended-range VLAN ID that is already assigned to an internal VLAN, an error message is generated, and the extended-range VLAN is rejected. To manually free an internal VLAN ID, you must temporarily shut down the routed port that is using the internal VLAN ID.



Note Devices running the LAN Base images support only static routing on SVIs.

SUMMARY STEPS

1. **enable**
2. **show vlan internal usage**
3. **configure terminal**
4. **interface *interface-id***
5. **shutdown**
6. **exit**
7. **vtp mode transparent**
8. **vlan *vlan-id***
9. **exit**
10. **interface *interface-id***
11. **no shutdown**
12. **end**
13. **copy running-config startup config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p>enable</p> <p>Example:</p>	<p>Enables privileged EXEC mode.</p> <ul style="list-style-type: none"> • Enter your password if prompted.

	Command or Action	Purpose
	Device> enable	
Step 2	show vlan internal usage Example: Device# show vlan internal usage	Displays the VLAN IDs being used internally by the device. If the VLAN ID that you want to use is an internal VLAN, the display shows the routed port that is using the VLAN ID. Enter that port number in Step 3.
Step 3	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 4	interface interface-id Example: Device(config)# interface gigabitethernet1/0/3	Specifies the interface ID for the routed port that is using the VLAN ID, and enters interface configuration mode.
Step 5	shutdown Example: Device(config-if)# shutdown	Shuts down the port to free the internal VLAN ID.
Step 6	exit Example: Device(config-if)# exit	Returns to global configuration mode.
Step 7	vtp mode transparent Example: Device(config)# vtp mode transparent	Sets the VTP mode to transparent for creating extended-range VLANs. Note This step is not required for VTP version 3.
Step 8	vlan vlan-id Example: Device(config-vlan)# vlan 2000	Enters the new extended-range VLAN ID, and enters VLAN configuration mode.
Step 9	exit Example:	Exits from VLAN configuration mode, and returns to global configuration mode.

	Command or Action	Purpose
	Device(config-vlan)# exit	
Step 10	interface <i>interface-id</i> Example: Device(config)# interface gigabitethernet1/0/3	Specifies the interface ID for the routed port that you shut down in Step 4, and enters interface configuration mode.
Step 11	no shutdown Example: Device(config)# no shutdown	Reenables the routed port. It will be assigned a new internal VLAN ID.
Step 12	end Example: Device(config)# end	Returns to privileged EXEC mode.
Step 13	copy running-config startup config Example: Device# copy running-config startup-config	Saves your entries in the device startup configuration file. To save an extended-range VLAN configuration, you need to save the VTP transparent mode configuration and the extended-range VLAN configuration in the device startup configuration file. Otherwise, if the device resets, it will default to VTP server mode, and the extended-range VLAN IDs will not be saved. Note This step is not required for VTP version 3 because VLANs are saved in the VLAN database.

Monitoring VLANs

Table 169: Privileged EXEC show Commands

Command	Purpose
show interfaces [vlan <i>vlan-id</i>]	Displays characteristics for all interfaces or for the specified VLAN configured on the device.

Configuration Examples

Example: Creating a VLAN Name

This example shows how to create Ethernet VLAN 20, name it test20, and add it to the VLAN database:

```
Switch# configure terminal
Switch(config)# vlan 20
Switch(config-vlan)# name test20
Switch(config-vlan)# end
```

Example: Configuring a Port as Access Port

This example shows how to configure a port as an access port in VLAN 2:

```
Switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Switch(config)# interface gigabitethernet 1/0/1
Switch(config-if)# switchport mode access
Switch(config-if)# switchport access vlan 2
Switch(config-if)# end
```

Example: Creating an Extended-Range VLAN

This example shows how to create a new extended-range VLAN with all default characteristics, enter VLAN configuration mode, and save the new VLAN in the switch startup configuration file:

```
Switch(config)# vtp mode transparent
Switch(config)# vlan 2000
Switch(config-vlan)# end
Switch# copy running-config startup config
```

Where to Go Next

After configuring VLANs, you can configure the following:

- VLAN Trunking Protocol (VTP)
- VLAN trunks
- Private VLANs
- Tunneling

Additional References

Related Documents

Related Topic	Document Title
For complete syntax and usage information for the commands used in this chapter.	<i>Catalyst 2960-XR Switch VLAN Management Command Reference</i>

Standards and RFCs

Standard/RFC	Title
—	—

MIBs

MIB	MIBs Link
All the supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
<p>The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies.</p> <p>To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds.</p> <p>Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.</p>	http://www.cisco.com/support

Feature History and Information for VLAN

Release	Modification
Cisco IOS Release 15.0(2)EX1	This feature was introduced.



CHAPTER 86

Configuring VLAN Trunks

- [Prerequisites for VLAN Trunks, on page 1719](#)
- [Restrictions for VLAN Trunks, on page 1719](#)
- [Information About VLAN Trunks, on page 1720](#)
- [How to Configure VLAN Trunks, on page 1724](#)
- [Configuration Examples for VLAN Trunking, on page 1737](#)
- [Where to Go Next, on page 1737](#)
- [Additional References, on page 1738](#)
- [Feature History and Information for VLAN Trunks, on page 1738](#)

Prerequisites for VLAN Trunks

The IEEE 802.1Q trunks impose these limitations on the trunking strategy for a network:

- In a network of Cisco devices connected through IEEE 802.1Q trunks, the devices maintain one spanning-tree instance for each VLAN allowed on the trunks. Non-Cisco devices might support one spanning-tree instance for all VLANs.

When you connect a Cisco device to a non-Cisco device through an IEEE 802.1Q trunk, the Cisco device combines the spanning-tree instance of the VLAN of the trunk with the spanning-tree instance of the non-Cisco IEEE 802.1Q device. However, spanning-tree information for each VLAN is maintained by Cisco devices separated by a cloud of non-Cisco IEEE 802.1Q devices. The non-Cisco IEEE 802.1Q cloud separating the Cisco devices is treated as a single trunk link between the devices.

- Make sure the native VLAN for an IEEE 802.1Q trunk is the same on both ends of the trunk link. If the native VLAN on one end of the trunk is different from the native VLAN on the other end, spanning-tree loops might result.
- Disabling spanning tree on the native VLAN of an IEEE 802.1Q trunk without disabling spanning tree on every VLAN in the network can potentially cause spanning-tree loops. We recommend that you leave spanning tree enabled on the native VLAN of an IEEE 802.1Q trunk or disable spanning tree on every VLAN in the network. Make sure your network is loop-free before disabling spanning tree.

Restrictions for VLAN Trunks

The following are restrictions for VLAN trunks:

- A trunk port cannot be a secure port.
- A trunk port cannot be a tunnel port.
- Trunk ports can be grouped into EtherChannel port groups, but all trunks in the group must have the same configuration. When a group is first created, all ports follow the parameters set for the first port to be added to the group. If you change the configuration of one of these parameters, the device propagates the setting that you entered to all ports in the group:
 - Allowed-VLAN list.
 - STP port priority for each VLAN.
 - STP Port Fast setting.
 - Trunk status:
 - If one port in a port group ceases to be a trunk, all ports cease to be trunks.
- We recommend that you configure no more than 24 trunk ports in Per VLAN Spanning Tree (PVST) mode and no more than 40 trunk ports in Multiple Spanning Tree (MST) mode.
- If you try to enable IEEE 802.1x on a trunk port, an error message appears, and IEEE 802.1x is not enabled. If you try to change the mode of an IEEE 802.1x-enabled port to trunk, the port mode is not changed.
- A port in dynamic mode can negotiate with its neighbor to become a trunk port. If you try to enable IEEE 802.1x on a dynamic port, an error message appears, and IEEE 802.1x is not enabled. If you try to change the mode of an IEEE 802.1x-enabled port to dynamic, the port mode is not changed.
- Dynamic Trunking Protocol (DTP) is not supported on tunnel ports.
- The device does not support Layer 3 trunks; you cannot configure subinterfaces or use the **encapsulation** keyword on Layer 3 interfaces. The device does support Layer 2 trunks and Layer 3 VLAN interfaces, which provide equivalent capabilities.

Information About VLAN Trunks

Trunking Overview

A trunk is a point-to-point link between one or more Ethernet device interfaces and another networking device such as a router or a device. Ethernet trunks carry the traffic of multiple VLANs over a single link, and you can extend the VLANs across an entire network.

The following trunking encapsulations are available on all Ethernet interfaces:

- IEEE 802.1Q— Industry-standard trunking encapsulation.

Trunking Modes

Ethernet trunk interfaces support different trunking modes. You can set an interface as trunking or nontrunking or to negotiate trunking with the neighboring interface. To autonegotiate trunking, the interfaces must be in the same VTP domain.

Trunk negotiation is managed by the Dynamic Trunking Protocol (DTP), which is a Point-to-Point Protocol (PPP). However, some internetworking devices might forward DTP frames improperly, which could cause misconfigurations.

To avoid this, you should configure interfaces connected to devices that do not support DTP to not forward DTP frames, that is, to turn off DTP.

- If you do not intend to trunk across those links, use the **switchport mode access** interface configuration command to disable trunking.

Layer 2 Interface Modes

Table 170: Layer 2 Interface Modes

Mode	Function
switchport mode access	Puts the interface (access port) into permanent nontrunking mode and negotiates to convert the link into a nontrunk link. The interface becomes a nontrunk interface regardless of whether or not the neighboring interface is a trunk interface.
switchport mode dynamic auto	Makes the interface able to convert the link to a trunk link. The interface becomes a trunk interface if the neighboring interface is set to trunk or desirable mode. The default switchport mode for all Ethernet interfaces is dynamic auto .
switchport mode dynamic desirable	Makes the interface actively attempt to convert the link to a trunk link. The interface becomes a trunk interface if the neighboring interface is set to trunk , desirable , or auto mode.
switchport mode trunk	Puts the interface into permanent trunking mode and negotiates to convert the neighboring link into a trunk link. The interface becomes a trunk interface even if the neighboring interface is not a trunk interface.
switchport nonegotiate	Prevents the interface from generating DTP frames. You can use this command only when the interface switchport mode is access or trunk . You must manually configure the neighboring interface as a trunk interface to establish a trunk link.
switchport mode dot1q-tunnel	Configures the interface as a tunnel (nontrunking) port to be connected in an asymmetric link with an IEEE 802.1Q trunk port. The IEEE 802.1Q tunneling is used to maintain customer VLAN integrity across a service provider network.

Mode	Function
switchport mode private-vlan	Configures the private VLAN mode. Note The switchport mode private-vlan command option is not supported.

Allowed VLANs on a Trunk

By default, a trunk port sends traffic to and receives traffic from all VLANs. All VLAN IDs, 1 to 4094, are allowed on each trunk. However, you can remove VLANs from the allowed list, preventing traffic from those VLANs from passing over the trunk.

To reduce the risk of spanning-tree loops or storms, you can disable VLAN 1 on any individual VLAN trunk port by removing VLAN 1 from the allowed list. When you remove VLAN 1 from a trunk port, the interface continues to send and receive management traffic, for example, Cisco Discovery Protocol (CDP), Port Aggregation Protocol (PAgP), Link Aggregation Control Protocol (LACP), DTP, and VTP in VLAN 1.

If a trunk port with VLAN 1 disabled is converted to a nontrunk port, it is added to the access VLAN. If the access VLAN is set to 1, the port will be added to VLAN 1, regardless of the **switchport trunk allowed** setting. The same is true for any VLAN that has been disabled on the port.

A trunk port can become a member of a VLAN if the VLAN is enabled, if VTP knows of the VLAN, and if the VLAN is in the allowed list for the port. When VTP detects a newly enabled VLAN and the VLAN is in the allowed list for a trunk port, the trunk port automatically becomes a member of the enabled VLAN. When VTP detects a new VLAN and the VLAN is not in the allowed list for a trunk port, the trunk port does not become a member of the new VLAN.

Load Sharing on Trunk Ports

Load sharing divides the bandwidth supplied by parallel trunks connecting devices. To avoid loops, STP normally blocks all but one parallel link between devices. Using load sharing, you divide the traffic between the links according to which VLAN the traffic belongs.

You configure load sharing on trunk ports by using STP port priorities or STP path costs. For load sharing using STP port priorities, both load-sharing links must be connected to the same device. For load sharing using STP path costs, each load-sharing link can be connected to the same device or to two different devices.

Network Load Sharing Using STP Priorities

When two ports on the same device form a loop, the device uses the STP port priority to decide which port is enabled and which port is in a blocking state. You can set the priorities on a parallel trunk port so that the port carries all the traffic for a given VLAN. The trunk port with the higher priority (lower values) for a VLAN is forwarding traffic for that VLAN. The trunk port with the lower priority (higher values) for the same VLAN remains in a blocking state for that VLAN. One trunk port sends or receives all traffic for the VLAN.

Network Load Sharing Using STP Path Cost

You can configure parallel trunks to share VLAN traffic by setting different path costs on a trunk and associating the path costs with different sets of VLANs, blocking different ports for different VLANs. The VLANs keep the traffic separate and maintain redundancy in the event of a lost link.

Feature Interactions

Trunking interacts with other features in these ways:

- A trunk port cannot be a secure port.
- A trunk port cannot be a tunnel port.
- Trunk ports can be grouped into EtherChannel port groups, but all trunks in the group must have the same configuration. When a group is first created, all ports follow the parameters set for the first port to be added to the group. If you change the configuration of one of these parameters, the device propagates the setting that you entered to all ports in the group:
 - Allowed-VLAN list.
 - STP port priority for each VLAN.
 - STP Port Fast setting.
 - Trunk status:

If one port in a port group ceases to be a trunk, all ports cease to be trunks.

- We recommend that you configure no more than 24 trunk ports in Per VLAN Spanning Tree (PVST) mode and no more than 40 trunk ports in Multiple Spanning Tree (MST) mode.
- If you try to enable IEEE 802.1x on a trunk port, an error message appears, and IEEE 802.1x is not enabled. If you try to change the mode of an IEEE 802.1x-enabled port to trunk, the port mode is not changed.
- A port in dynamic mode can negotiate with its neighbor to become a trunk port. If you try to enable IEEE 802.1x on a dynamic port, an error message appears, and IEEE 802.1x is not enabled. If you try to change the mode of an IEEE 802.1x-enabled port to dynamic, the port mode is not changed.

Default Layer 2 Ethernet Interface VLAN Configuration

The following table shows the default Layer 2 Ethernet interface VLAN configuration.

Table 171: Default Layer 2 Ethernet Interface VLAN Configuration

Feature	Default Setting
Interface mode	switchport mode dynamic auto
Trunk encapsulation	switchport trunk encapsulation negotiate
Allowed VLAN range	VLANs 1 to 4094
VLAN range eligible for pruning	VLANs 2 to 1001
Default VLAN (for access ports)	VLAN 1
Native VLAN (for IEEE 802.1Q trunks)	VLAN 1

How to Configure VLAN Trunks

To avoid trunking misconfigurations, configure interfaces connected to devices that do not support DTP to not forward DTP frames, that is, to turn off DTP.

- If you do not intend to trunk across those links, use the **switchport mode access** interface configuration command to disable trunking.
- To enable trunking to a device that does not support DTP, use the **switchport mode trunk** and **switchport nonegotiate** interface configuration commands to cause the interface to become a trunk but to not generate DTP frames.

Configuring an Ethernet Interface as a Trunk Port

Configuring a Trunk Port

Because trunk ports send and receive VTP advertisements, to use VTP you must ensure that at least one trunk port is configured on the device and that this trunk port is connected to the trunk port of a second device. Otherwise, the device cannot receive any VTP advertisements.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface *interface-id***
4. **switchport mode {dynamic {auto | desirable} | trunk}**
5. **switchport access vlan *vlan-id***
6. **switchport trunk native vlan *vlan-id***
7. **end**
8. **show interfaces *interface-id* switchport**
9. **show interfaces *interface-id* trunk**
10. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.

	Command or Action	Purpose
Step 3	interface <i>interface-id</i> Example: Device(config)# interface gigabitethernet 1/0/2	Specifies the port to be configured for trunking, and enters interface configuration mode.
Step 4	switchport mode {dynamic {auto desirable} trunk} Example: Device(config-if)# switchport mode dynamic desirable	Configures the interface as a Layer 2 trunk (required only if the interface is a Layer 2 access port or tunnel port or to specify the trunking mode). <ul style="list-style-type: none"> • dynamic auto—Sets the interface to a trunk link if the neighboring interface is set to trunk or desirable mode. This is the default. • dynamic desirable—Sets the interface to a trunk link if the neighboring interface is set to trunk, desirable, or auto mode. • trunk—Sets the interface in permanent trunking mode and negotiate to convert the link to a trunk link even if the neighboring interface is not a trunk interface.
Step 5	switchport access vlan <i>vlan-id</i> Example: Device(config-if)# switchport access vlan 200	(Optional) Specifies the default VLAN, which is used if the interface stops trunking.
Step 6	switchport trunk native vlan <i>vlan-id</i> Example: Device(config-if)# switchport trunk native vlan 200	Specifies the native VLAN for IEEE 802.1Q trunks.
Step 7	end Example: Device(config)# end	Returns to privileged EXEC mode.
Step 8	show interfaces <i>interface-id</i> switchport Example: Device# show interfaces gigabitethernet 1/0/2 switchport	Displays the switch port configuration of the interface in the <i>Administrative Mode</i> and the <i>Administrative Trunking Encapsulation</i> fields of the display.

	Command or Action	Purpose
Step 9	show interfaces <i>interface-id</i> trunk Example: <pre>Device# show interfaces gigabitethernet 1/0/2 trunk</pre>	Displays the trunk configuration of the interface.
Step 10	copy running-config startup-config Example: <pre>Device# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file. Note To return an interface to its default configuration, use the default interface <i>interface-id</i> interface configuration command. To reset all trunking characteristics of a trunking interface to the defaults, use the no switchport trunk interface configuration command. To disable trunking, use the switchport mode access interface configuration command to configure the port as a static-access port.

Defining the Allowed VLANs on a Trunk

VLAN 1 is the default VLAN on all trunk ports in all Cisco devices, and it has previously been a requirement that VLAN 1 always be enabled on every trunk link. You can use the VLAN 1 minimization feature to disable VLAN 1 on any individual VLAN trunk link so that no user traffic (including spanning-tree advertisements) is sent or received on VLAN 1.

SUMMARY STEPS

1. **configure terminal**
2. **interface *interface-id***
3. **switchport mode trunk**
4. **switchport trunk allowed vlan {add | all | except | none | remove} *vlan-list***
5. **end**
6. **show interfaces *interface-id* switchport**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>Device# configure terminal</pre>	Enters global configuration mode.

	Command or Action	Purpose
Step 2	interface <i>interface-id</i> Example: Device(config)# interface gigabitethernet1/0/1	Specifies the port to be configured, and enters interface configuration mode.
Step 3	switchport mode trunk Example: Device(config-if)# switchport mode trunk	Configures the interface as a VLAN trunk port.
Step 4	switchport trunk allowed vlan {add all except none remove} <i>vlan-list</i> Example: Device(config-if)# switchport trunk allowed vlan remove 2	(Optional) Configures the list of VLANs allowed on the trunk. The <i>vlan-list</i> parameter is either a single VLAN number from 1 to 4094 or a range of VLANs described by two VLAN numbers, the lower one first, separated by a hyphen. Do not enter any spaces between comma-separated VLAN parameters or in hyphen-specified ranges. All VLANs are allowed by default.
Step 5	end Example: Device(config)# end	Returns to privileged EXEC mode.
Step 6	show interfaces <i>interface-id</i> switchport Example: Device# show interfaces gigabitethernet1/0/1 switchport	Verifies your entries in the <i>Trunking VLANs Enabled</i> field of the display.
Step 7	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file. Note To return to the default allowed VLAN list of all VLANs, use the no switchport trunk allowed vlan interface configuration command.

Changing the Pruning-Eligible List

The pruning-eligible list applies only to trunk ports. Each trunk port has its own eligibility list. VTP pruning must be enabled for this procedure to take effect.

SUMMARY STEPS

1. enable

2. **configure terminal**
3. **interface** *interface-id*
4. **switchport trunk pruning vlan** {**add** | **except** | **none** | **remove**} *vlan-list* [,*vlan* [,*vlan* [,...]]]
5. **end**
6. **show interfaces** *interface-id* **switchport**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	interface <i>interface-id</i> Example: Device(config)# interface gigabitethernet0/1	Selects the trunk port for which VLANs should be pruned, and enters interface configuration mode.
Step 4	switchport trunk pruning vlan { add except none remove } <i>vlan-list</i> [, <i>vlan</i> [, <i>vlan</i> [,...]]]	Configures the list of VLANs allowed to be pruned from the trunk. For explanations about using the add , except , none , and remove keywords, see the command reference for this release. Separate non-consecutive VLAN IDs with a comma and no spaces; use a hyphen to designate a range of IDs. Valid IDs are 2 to 1001. Extended-range VLANs (VLAN IDs 1006 to 4094) cannot be pruned. VLANs that are pruning-ineligible receive flooded traffic. The default list of VLANs allowed to be pruned contains VLANs 2 to 1001. Note To return to the default pruning-eligible list of all VLANs, use the no switchport trunk pruning vlan interface configuration command.
Step 5	end Example:	Returns to privileged EXEC mode.

	Command or Action	Purpose
	Device(config)# end	
Step 6	show interfaces <i>interface-id</i> switchport Example: Device# show interfaces gigabitethernet 1/0/1 switchport	Verifies your entries in the <i>Pruning VLANs Enabled</i> field of the display.
Step 7	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Configuring the Native VLAN for Untagged Traffic

A trunk port configured with IEEE 802.1Q tagging can receive both tagged and untagged traffic. By default, the device forwards untagged traffic in the native VLAN configured for the port. The native VLAN is VLAN 1 by default.

The native VLAN can be assigned any VLAN ID.

If a packet has a VLAN ID that is the same as the outgoing port native VLAN ID, the packet is sent untagged; otherwise, the device sends the packet with a tag.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface *interface-id***
4. **switchport trunk native vlan *vlan-id***
5. **end**
6. **show interfaces *interface-id* switchport**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example:	Enters global configuration mode.

	Command or Action	Purpose
	Device# <code>configure terminal</code>	
Step 3	interface <i>interface-id</i> Example: Device(config)# <code>interface gigabitethernet 1/0/2</code>	Defines the interface that is configured as the IEEE 802.1Q trunk, and enters interface configuration mode.
Step 4	switchport trunk native vlan <i>vlan-id</i> Example: Device(config-if)# <code>switchport trunk native vlan 12</code>	Configures the VLAN that is sending and receiving untagged traffic on the trunk port. For <i>vlan-id</i> , the range is 1 to 4094. Note To return to the default native VLAN, VLAN 1, use the no switchport trunk native vlan interface configuration command.
Step 5	end Example: Device(config-if)# <code>end</code>	Returns to privileged EXEC mode.
Step 6	show interfaces <i>interface-id</i> switchport Example: Device# <code>show interfaces gigabitethernet 1/0/2 switchport</code>	Verifies your entries in the <i>Trunking Native Mode VLAN</i> field.
Step 7	copy running-config startup-config Example: Device# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Configuring Trunk Ports for Load Sharing

Configuring Load Sharing Using STP Port Priorities

These steps describe how to configure a network with load sharing using STP port priorities.

SUMMARY STEPS

1. `enable`
2. `configure terminal`
3. `vtp domain domain-name`

4. **vtp mode server**
5. **end**
6. **show vtp status**
7. **show vlan**
8. **configure terminal**
9. **interface *interface-id***
10. **switchport mode trunk**
11. **end**
12. **show interfaces *interface-id* switchport**
13. Repeat the above steps on Device A for a second port in the device.
14. Repeat the above steps on Device B to configure the trunk ports that connect to the trunk ports configured on Device A.
15. **show vlan**
16. **configure terminal**
17. **interface *interface-id***
18. **spanning-tree vlan *vlan-range* port-priority *priority-value***
19. **exit**
20. **interface *interface-id***
21. **spanning-tree vlan *vlan-range* port-priority *priority-value***
22. **end**
23. **show running-config**
24. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode on Device A.
Step 3	vtp domain <i>domain-name</i> Example: Device(config)# vtp domain workdomain	Configures a VTP administrative domain. The domain name can be 1 to 32 characters.
Step 4	vtp mode server Example:	Configures Device A as the VTP server.

	Command or Action	Purpose
	Device(config)# vtp mode server	
Step 5	end Example: Device(config)# end	Returns to privileged EXEC mode.
Step 6	show vtp status Example: Device# show vtp status	Verifies the VTP configuration on both Device A and Device B. In the display, check the <i>VTP Operating Mode</i> and the <i>VTP Domain Name</i> fields.
Step 7	show vlan Example: Device# show vlan	Verifies that the VLANs exist in the database on Device A.
Step 8	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 9	interface interface-id Example: Device(config)# interface gigabitethernet1/0/1	Defines the interface to be configured as a trunk, and enters interface configuration mode.
Step 10	switchport mode trunk Example: Device(config-if)# switchport mode trunk	Configures the port as a trunk port.
Step 11	end Example: Device(config-if)# end	Returns to privileged EXEC mode.
Step 12	show interfaces interface-id switchport Example: Device# show interfaces gigabitethernet 1/0/1	Verifies the VLAN configuration.

	Command or Action	Purpose
	<code>switchport</code>	
Step 13	Repeat the above steps on Device A for a second port in the device.	
Step 14	Repeat the above steps on Device B to configure the trunk ports that connect to the trunk ports configured on Device A.	
Step 15	show vlan Example: Device# <code>show vlan</code>	When the trunk links come up, VTP passes the VTP and VLAN information to Device B. This command verifies that Device B has learned the VLAN configuration.
Step 16	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode on Device A.
Step 17	interface interface-id Example: Device(config)# <code>interface gigabitethernet 1/0/1</code>	Defines the interface to set the STP port priority, and enters interface configuration mode.
Step 18	spanning-tree vlan vlan-range port-priority priority-value Example: Device(config-if)# <code>spanning-tree vlan 8-10 port-priority 16</code>	Assigns the port priority for the VLAN range specified. Enter a port priority value from 0 to 240. Port priority values increment by 16.
Step 19	exit Example: Device(config-if)# <code>exit</code>	Returns to global configuration mode.
Step 20	interface interface-id Example: Device(config)# <code>interface gigabitethernet 1/0/2</code>	Defines the interface to set the STP port priority, and enters interface configuration mode.
Step 21	spanning-tree vlan vlan-range port-priority priority-value Example:	Assigns the port priority for the VLAN range specified. Enter a port priority value from 0 to 240. Port priority values increment by 16.

	Command or Action	Purpose
	Device (config-if) # spanning-tree vlan 3-6 port-priority 16	
Step 22	end Example: Device (config-if) # end	Returns to privileged EXEC mode.
Step 23	show running-config Example: Device# show running-config	Verifies your entries.
Step 24	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Configuring Load Sharing Using STP Path Cost

These steps describe how to configure a network with load sharing using STP path costs.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **switchport mode trunk**
5. **exit**
6. Repeat Steps 2 through 4 on a second interface in Device A .
7. **end**
8. **show running-config**
9. **show vlan**
10. **configure terminal**
11. **interface** *interface-id*
12. **spanning-tree vlan** *vlan-range* **cost** *cost-value*
13. **end**
14. Repeat Steps 9 through 13 on the other configured trunk interface on Device A, and set the spanning-tree path cost to 30 for VLANs 8, 9, and 10.
15. **exit**
16. **show running-config**
17. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode on Device A.
Step 3	interface <i>interface-id</i> Example: Device(config)# interface gigabitethernet 1/0/1	Defines the interface to be configured as a trunk, and enters interface configuration mode.
Step 4	switchport mode trunk Example: Device(config-if)# switchport mode trunk	Configures the port as a trunk port.
Step 5	exit Example: Device(config-if)# exit	Returns to global configuration mode.
Step 6	Repeat Steps 2 through 4 on a second interface in Device A .	
Step 7	end Example: Device(config)# end	Returns to privileged EXEC mode.
Step 8	show running-config Example: Device# show running-config	Verifies your entries. In the display, make sure that the interfaces are configured as trunk ports.

	Command or Action	Purpose
Step 9	show vlan Example: Device# <code>show vlan</code>	When the trunk links come up, Device A receives the VTP information from the other devices. This command verifies that Device A has learned the VLAN configuration.
Step 10	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode.
Step 11	interface <i>interface-id</i> Example: Device(config)# <code>interface gigabitethernet 1/0/1</code>	Defines the interface on which to set the STP cost, and enters interface configuration mode.
Step 12	spanning-tree vlan <i>vlan-range</i> cost <i>cost-value</i> Example: Device(config-if)# <code>spanning-tree vlan 2-4 cost 30</code>	Sets the spanning-tree path cost to 30 for VLANs 2 through 4.
Step 13	end Example: Device(config-if)# <code>end</code>	Returns to global configuration mode.
Step 14	Repeat Steps 9 through 13 on the other configured trunk interface on Device A, and set the spanning-tree path cost to 30 for VLANs 8, 9, and 10.	
Step 15	exit Example: Device(config)# <code>exit</code>	Returns to privileged EXEC mode.
Step 16	show running-config Example: Device# <code>show running-config</code>	Verifies your entries. In the display, verify that the path costs are set correctly for both trunk interfaces.
Step 17	copy running-config startup-config Example:	(Optional) Saves your entries in the configuration file.

	Command or Action	Purpose
	Device# <code>copy running-config startup-config</code>	

Configuration Examples for VLAN Trunking

Example: Configuring an IEEE 802.1Q Trunk

This example shows how to configure a port as an IEEE 802.1Q trunk. The example assumes that the neighbor interface is configured to support IEEE 802.1Q trunking.

```
Switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Switch(config)# interface gigabitethernet0/2
Switch(config-if)# switchport mode dynamic desirable
Switch(config-if)# switchport trunk encapsulation dot1q
Switch(config-if)# end
```

Example: Removing a VLAN

This example shows how to remove VLAN 2 from the allowed VLAN list on a port:

```
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# switchport trunk allowed vlan remove 2
Switch(config-if)# end
```

Where to Go Next

After configuring VLAN trunks, you can configure the following:

- VTP
- VLANs
- Private VLANs
- VLAN Membership Policy Server (VMPS)
- Tunneling
- Voice VLANs

Additional References

Related Documents

Related Topic	Document Title
For complete syntax and usage information for the commands used in this chapter.	<i>Catalyst 2960-XR Switch VLAN Management Command Reference</i>

Standards and RFCs

Standard/RFC	Title
—	—

MIBs

MIB	MIBs Link
All the supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
<p>The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies.</p> <p>To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds.</p> <p>Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.</p>	http://www.cisco.com/support

Feature History and Information for VLAN Trunks

Release	Modification
Cisco IOS Release 15.0(2)EX1	This feature was introduced.



CHAPTER 87

Configuring Private VLANs

- [Prerequisites for Private VLANs, on page 1739](#)
- [Restrictions for Private VLANs, on page 1742](#)
- [Information About Private VLANs, on page 1743](#)
- [How to Configure Private VLANs, on page 1748](#)
- [Monitoring Private VLANs, on page 1757](#)
- [Configuration Examples for Private VLANs, on page 1757](#)
- [Where to Go Next, on page 1759](#)
- [Additional References, on page 1760](#)
- [Feature History and Information for Private VLANs, on page 1760](#)

Prerequisites for Private VLANs

The following are prerequisites for configuring private VLANs:

- When you configure private VLANs on switches running VTP, the switch must be in VTP transparent mode.
- When configuring private VLANs on the switch, always use the default Switch Database Management (SDM) template to balance system resources between unicast routes and Layer 2 entries. If another SDM template is configured, use the **sdm prefer default** global configuration command to set the default template.

Secondary and Primary VLAN Configuration

Follow these guidelines when configuring private VLANs:

- Private VLANs are supported in transparent mode for VTP 1, 2 and 3. If the device is running VTP version 1 or 2, you must set VTP to transparent mode. After you configure a private VLAN, you should not change the VTP mode to client or server. VTP version 3 supports private VLANs in all modes.
- With VTP version 1 or 2, after you have configured private VLANs, use the **copy running-config startup config** privileged EXEC command to save the VTP transparent mode configuration and private-VLAN configuration in the device startup configuration file. Otherwise, if the device resets, it defaults to VTP server mode, which does not support private VLANs. VTP version 3 does support private VLANs.

- VTP version 1 and 2 do not propagate private-VLAN configuration. You must configure private VLANs on each device where you want private-VLAN ports unless the devices are running VTP version 3, as VTP3 propagate private vlans.
- You cannot configure VLAN 1 or VLANs 1002 to 1005 as primary or secondary VLANs. Extended VLANs (VLAN IDs 1006 to 4094) can belong to private VLANs.
- A primary VLAN can have one isolated VLAN and multiple community VLANs associated with it. An isolated or community VLAN can have only one primary VLAN associated with it.
- Although a private VLAN contains more than one VLAN, only one Spanning Tree Protocol (STP) instance runs for the entire private VLAN. When a secondary VLAN is associated with the primary VLAN, the STP parameters of the primary VLAN are propagated to the secondary VLAN.
- When copying a PVLAN configuration from a tftp server and applying it on a running-config, the PVLAN association will not be formed. You will need to check and ensure that the primary VLAN is associated to all the secondary VLANs.

You can also use **configure replace flash:config_file force** instead of **copy flash:config_file running-config**.

- You can enable DHCP snooping on private VLANs. When you enable DHCP snooping on the primary VLAN, it is propagated to the secondary VLANs. If you configure DHCP on a secondary VLAN, the configuration does not take effect if the primary VLAN is already configured.
- When you enable IP source guard on private-VLAN ports, you must enable DHCP snooping on the primary VLAN.
- We recommend that you prune the private VLANs from the trunks on devices that carry no traffic in the private VLANs.
- You can apply different quality of service (QoS) configurations to primary, isolated, and community VLANs.
- Note the following considerations for sticky ARP:
 - Sticky ARP entries are those learned on SVIs and Layer 3 interfaces. These entries do not age out.
 - The **ip sticky-arp** global configuration command is supported only on SVIs belonging to private VLANs.
 - The **ip sticky-arp** interface configuration command is only supported on:
 - Layer 3 interfaces
 - SVIs belonging to normal VLANs
 - SVIs belonging to private VLANs

For more information about using the **ip sticky-arp global** configuration and the **ip sticky-arp interface** configuration commands, see the command reference for this release.

- You can configure VLAN maps on primary and secondary VLANs. However, we recommend that you configure the same VLAN maps on private-VLAN primary and secondary VLANs.
- PVLANS are bidirectional. They can be applied at both the ingress and egress sides.

When a frame in Layer-2 is forwarded within a private VLAN, the VLAN map is applied at the ingress side and at the egress side. When a frame is routed from inside a private VLAN to an external port, the private-VLAN map is applied at the ingress side. Similarly, when the frame is routed from an external port to a Private VLAN, the private-VLAN is applied at the egress side.

Bridging

- For upstream traffic from secondary VLAN to primary VLAN, the MAP of the secondary VLAN is applied on the ingress side and the MAP of the primary VLAN is applied on the egress side.
- For downstream traffic from primary VLAN to secondary VLAN, the MAP of the primary VLAN is applied in the ingress direction and the MAP of the secondary VLAN is applied in the egress direction.

Routing

If we have two private VLAN domains - PV1 (sec1, prim1) and PV2 (sec2, prim2). For frames routed from PV1 to PV2:

- The MAP of sec1 and L3 ACL of prim1 is applied in the ingress port .
- The MAP of sec1 and L3 ACL of prim2 is applied in the egress port.
- For packets going upstream or downstream from isolated host port to promiscuous port, the isolated VLAN's VACL is applied in the ingress direction and primary VLAN'S VACL is applied in the egress direction. This allows user to configure different VACL for different secondary VLAN in a same primary VLAN domain.

To filter out specific IP traffic for a private VLAN, you should apply the VLAN map to both the primary and secondary VLANs.

- You can apply router ACLs only on the primary-VLAN SVIs. The ACL is applied to both primary and secondary VLAN Layer 3 traffic.
- Although private VLANs provide host isolation at Layer 2, hosts can communicate with each other at Layer 3.
- Private VLANs support these Switched Port Analyzer (SPAN) features:
 - You can configure a private-VLAN port as a SPAN source port.
 - You can use VLAN-based SPAN (VSPAN) on primary, isolated, and community VLANs or use SPAN on only one VLAN to separately monitor egress or ingress traffic.

Private VLAN Port Configuration

Follow these guidelines when configuring private VLAN ports:

- Use only the private VLAN configuration commands to assign ports to primary, isolated, or community VLANs. Layer 2 access ports assigned to the VLANs that you configure as primary, isolated, or community VLANs are inactive while the VLAN is part of the private VLAN configuration. Layer 2 trunk interfaces remain in the STP forwarding state.
- Do not configure ports that belong to a PAgP or LACP EtherChannel as private VLAN ports. While a port is part of the private VLAN configuration, any EtherChannel configuration for it is inactive.

- Enable Port Fast and BPDU guard on isolated and community host ports to prevent STP loops due to misconfigurations and to speed up STP convergence. When enabled, STP applies the BPDU guard feature to all Port Fast-configured Layer 2 LAN ports. Do not enable Port Fast and BPDU guard on promiscuous ports.
- If you delete a VLAN used in the private VLAN configuration, the private VLAN ports associated with the VLAN become inactive.
- Private VLAN ports can be on different network devices if the devices are trunk-connected and the primary and secondary VLANs have not been removed from the trunk.

Restrictions for Private VLANs

The following are restrictions for configuring private VLANs:

- Private VLANs are only supported on switches running the IP Lite image.

Limitations with Other Features

When configuring private VLANs, remember these limitations with other features:



Note In some cases, the configuration is accepted with no error messages, but the commands have no effect.

- Do not configure fallback bridging on switches with private VLANs.
- When IGMP snooping is enabled on the switch (the default), the switch or switch stack supports no more than 20 private VLAN domains.
- Do not configure a remote SPAN (RSPAN) VLAN as a private VLAN primary or secondary VLAN.
- Do not configure private VLAN ports on interfaces configured for these other features:
 - Dynamic-access port VLAN membership
 - Dynamic Trunking Protocol (DTP)
 - Port Aggregation Protocol (PAgP)
 - Link Aggregation Control Protocol (LACP)
 - Multicast VLAN Registration (MVR)
 - Voice VLAN
 - Web Cache Communication Protocol (WCCP)
- You can configure IEEE 802.1x port-based authentication on a private VLAN port, but do not configure 802.1x with port security, voice VLAN, or per-user ACL on private VLAN ports.
- A private VLAN host or promiscuous port cannot be a SPAN destination port. If you configure a SPAN destination port as a private VLAN port, the port becomes inactive.

- If you configure a static MAC address on a promiscuous port in the primary VLAN, you must add the same static address to all associated secondary VLANs. If you configure a static MAC address on a host port in a secondary VLAN, you must add the same static MAC address to the associated primary VLAN. When you delete a static MAC address from a private VLAN port, you must remove all instances of the configured MAC address from the private VLAN.



Note Dynamic MAC addresses learned in one VLAN of a private VLAN are replicated in the associated VLANs. For example, a MAC address learned in a secondary VLAN is replicated in the primary VLAN. When the original dynamic MAC address is deleted or aged out, the replicated addresses are removed from the MAC address table.

- Configure Layer 3 VLAN interfaces (SVIs) only for primary VLANs.

Information About Private VLANs

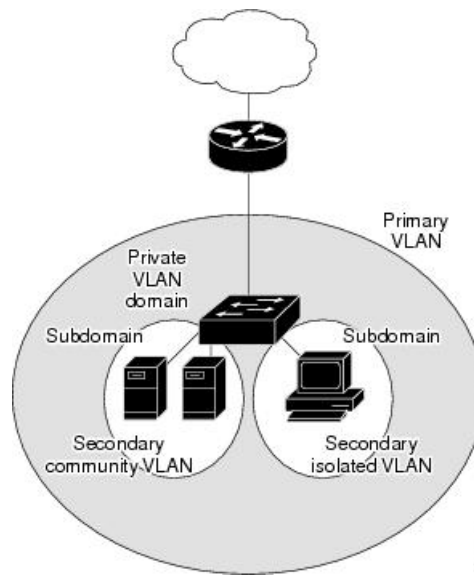
Private VLAN Domains

The private VLAN feature addresses two problems that service providers face when using VLANs:

- To enable IP routing, each VLAN is assigned a subnet address space or a block of addresses, which can result in wasting the unused IP addresses, and cause IP address management problems.

Figure 128: Private VLAN Domain

Using private VLANs addresses the scalability problem and provides IP address management benefits for service providers and Layer 2 security for customers. Private VLANs partition a regular VLAN domain into subdomains. A subdomain is represented by a pair of VLANs: a *primary* VLAN and a *secondary* VLAN. A private VLAN can have multiple VLAN pairs, one pair for each subdomain. All VLAN pairs in a private VLAN share the same primary VLAN. The secondary VLAN ID differentiates one subdomain from another.



Secondary VLANs

There are two types of secondary VLANs:

- Isolated VLANs—Ports within an isolated VLAN cannot communicate with each other at the Layer 2 level.
- Community VLANs—Ports within a community VLAN can communicate with each other but cannot communicate with ports in other communities at the Layer 2 level.

Private VLANs Ports

Private VLANs provide Layer 2 isolation between ports within the same private VLAN. Private VLAN ports are access ports that are one of these types:

- Promiscuous—A promiscuous port belongs to the primary VLAN and can communicate with all interfaces, including the community and isolated host ports that belong to the secondary VLANs associated with the primary VLAN.
- Isolated—An isolated port is a host port that belongs to an isolated secondary VLAN. It has complete Layer 2 separation from other ports within the same private VLAN, except for the promiscuous ports. Private VLANs block all traffic to isolated ports except traffic from promiscuous ports. Traffic received from an isolated port is forwarded only to promiscuous ports.
- Community—A community port is a host port that belongs to a community secondary VLAN. Community ports communicate with other ports in the same community VLAN and with promiscuous ports. These interfaces are isolated at Layer 2 from all other interfaces in other communities and from isolated ports within their private VLAN.



Note Trunk ports carry traffic from regular VLANs and also from primary, isolated, and community VLANs.

Primary and secondary VLANs have these characteristics:

- **Primary VLAN**—A private VLAN has only one primary VLAN. Every port in a private VLAN is a member of the primary VLAN. The primary VLAN carries unidirectional traffic downstream from the promiscuous ports to the (isolated and community) host ports and to other promiscuous ports.
- **Isolated VLAN**—A private VLAN has only one isolated VLAN. An isolated VLAN is a secondary VLAN that carries unidirectional traffic upstream from the hosts toward the promiscuous ports and the gateway.
- **Community VLAN**—A community VLAN is a secondary VLAN that carries upstream traffic from the community ports to the promiscuous port gateways and to other host ports in the same community. You can configure multiple community VLANs in a private VLAN.

A promiscuous port can serve only one primary VLAN, one isolated VLAN, and multiple community VLANs. Layer 3 gateways are typically connected to the device through a promiscuous port. With a promiscuous port, you can connect a wide range of devices as access points to a private VLAN. For example, you can use a promiscuous port to monitor or back up all the private VLAN servers from an administration workstation.

Private VLANs in Networks

In a switched environment, you can assign an individual private VLAN and associated IP subnet to each individual or common group of end stations. The end stations need to communicate only with a default gateway to communicate outside the private VLAN.

You can use private VLANs to control access to end stations in these ways:

- Configure selected interfaces connected to end stations as isolated ports to prevent any communication at Layer 2. For example, if the end stations are servers, this configuration prevents Layer 2 communication between the servers.
- Configure interfaces connected to default gateways and selected end stations (for example, backup servers) as promiscuous ports to allow all end stations access to a default gateway.

You can extend private VLANs across multiple devices by trunking the primary, isolated, and community VLANs to other devices that support private VLANs. To maintain the security of your private VLAN configuration and to avoid other use of the VLANs configured as private VLANs, configure private VLANs on all intermediate devices, including devices that have no private VLAN ports.

IP Addressing Scheme with Private VLANs

Assigning a separate VLAN to each customer creates an inefficient IP addressing scheme:

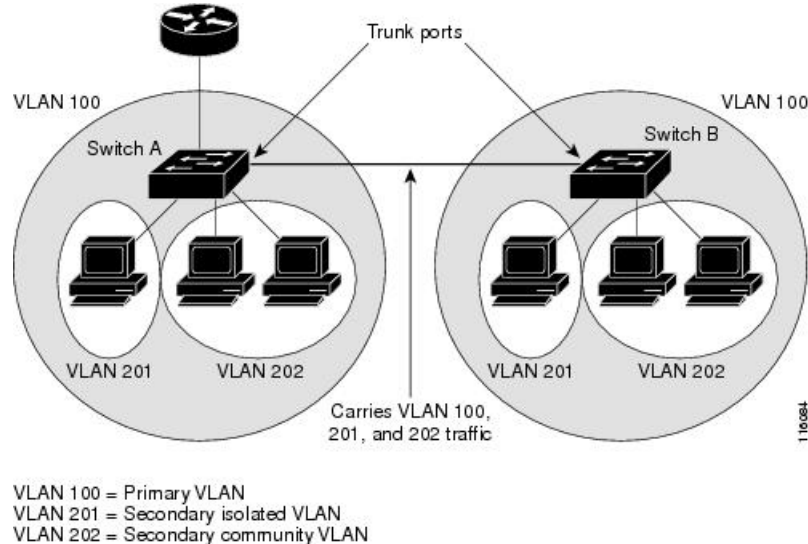
- Assigning a block of addresses to a customer VLAN can result in unused IP addresses.
- If the number of devices in the VLAN increases, the number of assigned address might not be large enough to accommodate them.

These problems are reduced by using private VLANs, where all members in the private VLAN share a common address space, which is allocated to the primary VLAN. Hosts are connected to secondary VLANs, and the DHCP server assigns them IP addresses from the block of addresses allocated to the primary VLAN. Subsequent IP addresses can be assigned to customer devices in different secondary VLANs, but in the same primary VLAN. When new devices are added, the DHCP server assigns them the next available address from a large pool of subnet addresses.

Private VLANs Across Multiple Devices

Figure 129: Private VLANs Across Switches

As with regular VLANs, private VLANs can span multiple devices. A trunk port carries the primary VLAN and secondary VLANs to a neighboring device. The trunk port treats the private VLAN as any other VLAN. A feature of private VLANs across multiple devices is that traffic from an isolated port in Device A does not reach an isolated port on Device B.



Private VLANs are supported in transparent mode for VTP 1, 2 and 3. Private vlan is also supported on server mode for VTP 3. If we have a server client setup using VTP 3, private vlans configured on the server should be reflected on the client.

Private VLAN Interaction with Other Features

Private VLANs and Unicast, Broadcast, and Multicast Traffic

In regular VLANs, devices in the same VLAN can communicate with each other at the Layer 2 level, but devices connected to interfaces in different VLANs must communicate at the Layer 3 level. In private VLANs, the promiscuous ports are members of the primary VLAN, while the host ports belong to secondary VLANs. Because the secondary VLAN is associated to the primary VLAN, members of these VLANs can communicate with each other at the Layer 2 level.

In a regular VLAN, broadcasts are forwarded to all ports in that VLAN. Private VLAN broadcast forwarding depends on the port sending the broadcast:

- An isolated port sends a broadcast only to the promiscuous ports or trunk ports.
- A community port sends a broadcast to all promiscuous ports, trunk ports, and ports in the same community VLAN.
- A promiscuous port sends a broadcast to all ports in the private VLAN (other promiscuous ports, trunk ports, isolated ports, and community ports).

Multicast traffic is routed or bridged across private VLAN boundaries and within a single community VLAN. Multicast traffic is not forwarded between ports in the same isolated VLAN or between ports in different secondary VLANs.

Private VLAN multicast forwarding supports the following:

- Sender can be outside the VLAN and the Receivers can be inside the VLAN domain.
- Sender can be inside the VLAN and the Receivers can be outside the VLAN domain.
- Sender and Receiver can both be in the same community vlan.

Private VLANs and SVIs

In a Layer 3 device, a device virtual interface (SVI) represents the Layer 3 interface of a VLAN. Layer 3 devices communicate with a private VLAN only through the primary VLAN and not through secondary VLANs. Configure Layer 3 VLAN interfaces (SVIs) only for primary VLANs. You cannot configure Layer 3 VLAN interfaces for secondary VLANs. SVIs for secondary VLANs are inactive while the VLAN is configured as a secondary VLAN.

- If you try to configure a VLAN with an active SVI as a secondary VLAN, the configuration is not allowed until you disable the SVI.
- If you try to create an SVI on a VLAN that is configured as a secondary VLAN and the secondary VLAN is already mapped at Layer 3, the SVI is not created, and an error is returned. If the SVI is not mapped at Layer 3, the SVI is created, but it is automatically shut down.

When the primary VLAN is associated with and mapped to the secondary VLAN, any configuration on the primary VLAN is propagated to the secondary VLAN SVIs. For example, if you assign an IP subnet to the primary VLAN SVI, this subnet is the IP subnet address of the entire private VLAN.

Private VLANs and Device Stacks

Private VLANs can operate within the device stack, and private-VLAN ports can reside on different stack members. However, the following changes to the stack can impact private-VLAN operation:

- If a stack contains only one private-VLAN promiscuous port and the stack member that contains that port is removed from the stack, host ports in that private VLAN lose connectivity outside the private VLAN.
- If a stack's active switch that contains the only private-VLAN promiscuous port in the stack fails or leaves the stack and a new active switch is elected, host ports in a private VLAN that had its promiscuous port on the old active switch lose connectivity outside of the private VLAN.
- If two stacks merge, private VLANs on the winning stack are not affected, but private-VLAN configuration on the losing device is lost when that device reboots.

Private VLAN Configuration Tasks

To configure a private VLAN, perform these steps:

1. Set VTP mode to transparent.
2. Create the primary and secondary VLANs and associate them.



Note If the VLAN is not created already, the private VLAN configuration process creates it.

3. Configure interfaces to be isolated or community host ports, and assign VLAN membership to the host port.
4. Configure interfaces as promiscuous ports, and map the promiscuous ports to the primary-secondary VLAN pair.
5. If inter-VLAN routing will be used, configure the primary SVI, and map the secondary VLANs to the primary.
6. Verify the private VLAN configuration.

Default Private VLAN Configuration

No private VLANs are configured.

How to Configure Private VLANs

Configuring and Associating VLANs in a Private VLAN

The **private-vlan** commands do not take effect until you exit VLAN configuration mode.

To configure and associate VLANs in a Private VLAN, perform these steps:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **vtp mode transparent**
4. **vlan *vlan-id***
5. **private-vlan primary**
6. **exit**
7. **vlan *vlan-id***
8. **private-vlan isolated**
9. **exit**
10. **vlan *vlan-id***
11. **private-vlan community**
12. **exit**
13. **vlan *vlan-id***
14. **private-vlan community**
15. **exit**
16. **vlan *vlan-id***
17. **private-vlan association [add | remove] *secondary_vlan_list***

18. `end`
19. `show vlan private-vlan [type]` or `show interfaces status`
20. `copy running-config startup config`

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> <code>enable</code>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode.
Step 3	vtp mode transparent Example: Device(config)# <code>vtp mode transparent</code>	Sets VTP mode to transparent (disable VTP). Note For VTP3, you can set mode to either server or transparent mode
Step 4	vlan <i>vlan-id</i> Example: Device(config)# <code>vlan 20</code>	Enters VLAN configuration mode and designates or creates a VLAN that will be the primary VLAN. The VLAN ID range is 2 to 1001 and 1006 to 4094.
Step 5	private-vlan primary Example: Device(config-vlan)# <code>private-vlan primary</code>	Designates the VLAN as the primary VLAN.
Step 6	exit Example: Device(config-vlan)# <code>exit</code>	Returns to global configuration mode.
Step 7	vlan <i>vlan-id</i> Example: Device(config)# <code>vlan 501</code>	(Optional) Enters VLAN configuration mode and designates or creates a VLAN that will be an isolated VLAN. The VLAN ID range is 2 to 1001 and 1006 to 4094.

	Command or Action	Purpose
Step 8	private-vlan isolated Example: Device(config-vlan)# private-vlan isolated	Designates the VLAN as an isolated VLAN.
Step 9	exit Example: Device(config-vlan)# exit	Returns to global configuration mode.
Step 10	vlan <i>vlan-id</i> Example: Device(config)# vlan 502	(Optional) Enters VLAN configuration mode and designates or creates a VLAN that will be a community VLAN. The VLAN ID range is 2 to 1001 and 1006 to 4094.
Step 11	private-vlan community Example: Device(config-vlan)# private-vlan community	Designates the VLAN as a community VLAN.
Step 12	exit Example: Device(config-vlan)# exit	Returns to global configuration mode.
Step 13	vlan <i>vlan-id</i> Example: Device(config)# vlan 503	(Optional) Enters VLAN configuration mode and designates or creates a VLAN that will be a community VLAN. The VLAN ID range is 2 to 1001 and 1006 to 4094.
Step 14	private-vlan community Example: Device(config-vlan)# private-vlan community	Designates the VLAN as a community VLAN.
Step 15	exit Example: Device(config-vlan)# exit	Returns to global configuration mode.

	Command or Action	Purpose
Step 16	<p>vlan <i>vlan-id</i></p> <p>Example:</p> <pre>Device(config)# vlan 20</pre>	Enters VLAN configuration mode for the primary VLAN designated in Step 4.
Step 17	<p>private-vlan association [add remove] <i>secondary_vlan_list</i></p> <p>Example:</p> <pre>Device(config-vlan)# private-vlan association 501-503</pre>	<p>Associates the secondary VLANs with the primary VLAN. It can be a single private-VLAN ID or a hyphenated range of private-VLAN IDs.</p> <ul style="list-style-type: none"> • The <i>secondary_vlan_list</i> parameter cannot contain spaces. It can contain multiple comma-separated items. Each item can be a single private-VLAN ID or a hyphenated range of private-VLAN IDs. • The <i>secondary_vlan_list</i> parameter can contain multiple community VLAN IDs but only one isolated VLAN ID. • Enter a <i>secondary_vlan_list</i>, or use the add keyword with a <i>secondary_vlan_list</i> to associate secondary VLANs with a primary VLAN. • Use the remove keyword with a <i>secondary_vlan_list</i> to clear the association between secondary VLANs and a primary VLAN. • The command does not take effect until you exit VLAN configuration mode.
Step 18	<p>end</p> <p>Example:</p> <pre>Device(config)# end</pre>	Returns to privileged EXEC mode.
Step 19	<p>show vlan private-vlan [type] or show interfaces status</p> <p>Example:</p> <pre>Device# show vlan private-vlan</pre>	Verifies the configuration.
Step 20	<p>copy running-config startup config</p> <p>Example:</p> <pre>Device# copy running-config startup-config</pre>	Saves your entries in the device startup configuration file.

Configuring a Layer 2 Interface as a Private VLAN Host Port

Follow these steps to configure a Layer 2 interface as a private-VLAN host port and to associate it with primary and secondary VLANs:



Note Isolated and community VLANs are both secondary VLANs.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **switchport mode private-vlan host**
5. **switchport private-vlan host-association** *primary_vlan_id secondary_vlan_id*
6. **end**
7. **show interfaces** [*interface-id*] **switchport**
8. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	interface <i>interface-id</i> Example: Device(config)# interface gigabitethernet1/0/22	Enters interface configuration mode for the Layer 2 interface to be configured.
Step 4	switchport mode private-vlan host Example: Device(config-if)# switchport mode private-vlan host	Configures the Layer 2 port as a private-VLAN host port.

	Command or Action	Purpose
Step 5	<p>switchport private-vlan host-association <i>primary_vlan_id</i> <i>secondary_vlan_id</i></p> <p>Example:</p> <pre>Device(config-if)# switchport private-vlan host-association 20 501</pre>	<p>Associates the Layer 2 port with a private VLAN.</p> <p>Note This is a required step to associate the PVLAN to a Layer 2 interface.</p>
Step 6	<p>end</p> <p>Example:</p> <pre>Device(config)# end</pre>	Returns to privileged EXEC mode.
Step 7	<p>show interfaces [<i>interface-id</i>] switchport</p> <p>Example:</p> <pre>Device# show interfaces gigabitethernet1/0/22 switchport</pre>	Verifies the configuration.
Step 8	<p>copy running-config startup-config</p> <p>Example:</p> <pre>Device# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Configuring a Layer 2 Interface as a Private VLAN Promiscuous Port

Follow these steps to configure a Layer 2 interface as a private VLAN promiscuous port and map it to primary and secondary VLANs:



Note Isolated and community VLANs are both secondary VLANs.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **switchport mode private-vlan promiscuous**
5. **switchport private-vlan mapping** *primary_vlan_id* {**add** | **remove**} *secondary_vlan_list*
6. **end**
7. **show interfaces** [*interface-id*] **switchport**
8. **copy running-config startup config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	interface <i>interface-id</i> Example: Device(config)# interface gigabitethernet1/0/2	Enters interface configuration mode for the Layer 2 interface to be configured.
Step 4	switchport mode private-vlan promiscuous Example: Device(config-if)# switchport mode private-vlan promiscuous	Configures the Layer 2 port as a private VLAN promiscuous port.
Step 5	switchport private-vlan mapping <i>primary_vlan_id</i> { add remove } <i>secondary_vlan_list</i> Example: Device(config-if)# switchport private-vlan mapping 20 add 501-503	Maps the private VLAN promiscuous port to a primary VLAN and to selected secondary VLANs. <ul style="list-style-type: none"> • The <i>secondary_vlan_list</i> parameter cannot contain spaces. It can contain multiple comma-separated items. Each item can be a single private VLAN ID or a hyphenated range of private VLAN IDs. • Enter a <i>secondary_vlan_list</i>, or use the add keyword with a <i>secondary_vlan_list</i> to map the secondary VLANs to the private VLAN promiscuous port. • Use the remove keyword with a <i>secondary_vlan_list</i> to clear the mapping between secondary VLANs and the private VLAN promiscuous port.
Step 6	end Example: Device(config)# end	Returns to privileged EXEC mode.

	Command or Action	Purpose
Step 7	show interfaces [<i>interface-id</i>] switchport Example: <pre>Device# show interfaces gigabitethernet1/0/2 switchport</pre>	Verifies the configuration.
Step 8	copy running-config startup config Example: <pre>Device# copy running-config startup-config</pre>	Saves your entries in the device startup configuration file.

Mapping Secondary VLANs to a Primary VLAN Layer 3 VLAN Interface

If the private VLAN will be used for inter-VLAN routing, you configure an SVI for the primary VLAN and map secondary VLANs to the SVI.



Note Isolated and community VLANs are both secondary VLANs.

Follow these steps to map secondary VLANs to the SVI of a primary VLAN to allow Layer 3 switching of private VLAN traffic:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface vlan** *primary_vlan_id*
4. **private-vlan mapping** [**add** | **remove**] *secondary_vlan_list*
5. **end**
6. **show interface private-vlan mapping**
7. **copy running-config startup config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Device> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example:	Enters global configuration mode.

	Command or Action	Purpose
	Device# <code>configure terminal</code>	
Step 3	interface vlan <i>primary_vlan_id</i> Example: Device(config)# <code>interface vlan 20</code>	Enters interface configuration mode for the primary VLAN, and configures the VLAN as an SVI. The VLAN ID range is 2 to 1001 and 1006 to 4094.
Step 4	private-vlan mapping [add remove] <i>secondary_vlan_list</i> Example: Device(config-if)# <code>private-vlan mapping 501-503</code>	Maps the secondary VLANs to the Layer 3 VLAN interface of a primary VLAN to allow Layer 3 switching of private VLAN ingress traffic. Note The private-vlan mapping interface configuration command only affects private VLAN traffic that is Layer 3 switched. <ul style="list-style-type: none"> • The <i>secondary_vlan_list</i> parameter cannot contain spaces. It can contain multiple comma-separated items. Each item can be a single private-VLAN ID or a hyphenated range of private-VLAN IDs. • Enter a <i>secondary_vlan_list</i>, or use the add keyword with a <i>secondary_vlan_list</i> to map the secondary VLANs to a primary VLAN. • Use the remove keyword with a <i>secondary_vlan_list</i> to clear the mapping between secondary VLANs and a primary VLAN.
Step 5	end Example: Device(config)# <code>end</code>	Returns to privileged EXEC mode.
Step 6	show interface private-vlan mapping Example: Device# <code>show interfaces private-vlan mapping</code>	Verifies the configuration.
Step 7	copy running-config startup config Example: Device# <code>copy running-config startup-config</code>	Saves your entries in the device startup configuration file.

Monitoring Private VLANs

The following table displays the commands used to monitor private VLANs.

Table 172: Private VLAN Monitoring Commands

Command	Purpose
show interfaces status	Displays the status of interfaces, including the VLANs to which they belongs.
show vlan private-vlan [type]	Displays the private VLAN information for the Device.
show interface switchport	Displays private VLAN configuration on interfaces.
show interface private-vlan mapping	Displays information about the private VLAN mapping for VLAN SVIs.

Configuration Examples for Private VLANs

Example: Configuring a Primary VLAN, Isolated VLAN, and a Community of VLANs

This example shows how to configure VLAN 20 as a primary VLAN, VLAN 501 as an isolated VLAN, and VLANs 502 and 503 as community VLANs, to associate them in a private VLAN, and to verify the configuration:

```
Switch# configure terminal
Switch(config)# vlan 20
Switch(config-vlan)# private-vlan primary
Switch(config-vlan)# exit
Switch(config)# vlan 501
Switch(config-vlan)# private-vlan isolated
Switch(config-vlan)# exit
Switch(config)# vlan 502
Switch(config-vlan)# private-vlan community
Switch(config-vlan)# exit
Switch(config)# vlan 503
Switch(config-vlan)# private-vlan community
Switch(config-vlan)# exit
Switch(config)# vlan 20
Switch(config-vlan)# private-vlan association 501-503
Switch(config-vlan)# end
Switch(config)# show vlan private vlan
```

```
Primary Secondary Type Ports
-----
```

```
20 501 isolated
20 502 community
20 503 community
```

```
20 504 non-operational
```

Example: Configuring an Interface as a Host Port

This example shows how to configure an interface as a private VLAN host port, associate it with a private VLAN pair, and verify the configuration:

```
Device# configure terminal
Device(config)# interface gigabitethernet1/0/22
Device(config-if)# switchport mode private-vlan host
Device(config-if)# switchport private-vlan host-association 20 501
Device(config-if)# end
Device# show interfaces gigabitethernet1/0/22 switchport
Name: Gi1/0/22
Switchport: Enabled
Administrative Mode: private-vlan host
Operational Mode: private-vlan host
Administrative Trunking Encapsulation: negotiate
Operational Trunking Encapsulation: native
Negotiation of Trunking: Off
Access Mode VLAN: 1 (default)
Trunking Native Mode VLAN: 1 (default)
Administrative Native VLAN tagging: enabled
Voice VLAN: none
Administrative private-vlan host-association: 20 501
Administrative private-vlan mapping: none
Administrative private-vlan trunk native VLAN: none
Administrative private-vlan trunk Native VLAN tagging: enabled
Administrative private-vlan trunk encapsulation: dot1q
Administrative private-vlan trunk normal VLANs: none
Administrative private-vlan trunk private VLANs: none
Operational private-vlan:
20 501

<output truncated>
```

Example: Configuring an Interface as a Private VLAN Promiscuous Port

This example shows how to configure an interface as a private VLAN promiscuous port and map it to a private VLAN. The interface is a member of primary VLAN 20 and secondary VLANs 501 to 503 are mapped to it.

```
Device# configure terminal
Device(config)# interface gigabitethernet1/0/2
Device(config-if)# switchport mode private-vlan promiscuous
Device(config-if)# switchport private-vlan mapping 20 add 501-503
Device(config-if)# end
```

Use the **show vlan private-vlan** or the **show interface status** privileged EXEC command to display primary and secondary VLANs and private-VLAN ports on the Device.

Example: Mapping Secondary VLANs to a Primary VLAN Interface

This example shows how to map the interfaces for VLANs 501 and 502 to primary VLAN 10, which permits routing of secondary VLAN ingress traffic from private VLANs 501 and 502:

```
Device# configure terminal
Device(config)# interface vlan 20
Device(config-if)# private-vlan mapping 501-503
Device(config-if)# end
Device# show interfaces private-vlan mapping
Interface Secondary VLAN Type
-----
vlan20      501          isolated
vlan20      502          community
vlan20      503          community
```

Example: Monitoring Private VLANs

This example shows output from the `show vlan private-vlan` command:

```
Device# show vlan private-vlan
Primary Secondary Type Ports
-----
20      501      isolated  Gi1/0/22, Gi1/0/2
20      502      community Gi1/0/2
20      503      community Gi1/0/2
```

Where to Go Next

You can configure the following:

- VTP
- VLANs
- VLAN trunking
- VLAN Membership Policy Server (VMPS)
- Tunneling
- Voice VLANs

Additional References

Related Documents

Related Topic	Document Title
For complete syntax and usage information for the commands used in this chapter.	<i>Catalyst 2960-XR Switch VLAN Management Command Reference</i>

Standards and RFCs

Standard/RFC	Title
—	—

MIBs

MIB	MIBs Link
All the supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
<p>The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies.</p> <p>To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds.</p> <p>Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.</p>	http://www.cisco.com/support

Feature History and Information for Private VLANs

Release	Modification
Cisco IOS Release 15.0(2)EX1	This feature was introduced.



CHAPTER 88

Configuring VMPS

- [Prerequisites for VMPS, on page 1761](#)
- [Restrictions for VMPS, on page 1761](#)
- [Information About VMPS, on page 1762](#)
- [How to Configure VMPS, on page 1764](#)
- [Monitoring the VMPS, on page 1770](#)
- [Configuration Example for VMPS, on page 1771](#)
- [Where to Go Next, on page 1772](#)
- [Additional References, on page 1773](#)
- [Feature History and Information for VMPS, on page 1774](#)

Prerequisites for VMPS

- You should configure the VMPS before you configure ports as dynamic-access ports.
- When you configure a port as a dynamic-access port, the spanning-tree Port Fast feature is automatically enabled for that port. The Port Fast mode accelerates the process of bringing the port into the forwarding state.

Restrictions for VMPS

- IEEE 802.1x ports cannot be configured as dynamic-access ports. If you try to enable IEEE 802.1x on a dynamic-access (VQP) port, an error message appears, and IEEE 802.1x is not enabled. If you try to change an IEEE 802.1x-enabled port to dynamic VLAN assignment, an error message appears, and the VLAN configuration is not changed.
- Trunk ports cannot be dynamic-access ports, but you can enter the **switchport access vlan dynamic** interface configuration command for a trunk port. In this case, the switch retains the setting and applies it if the port is later configured as an access port.
You must turn off trunking on the port before the dynamic-access setting takes effect.
- Dynamic-access ports cannot be monitor ports.
- Secure ports cannot be dynamic-access ports. You must disable port security on a port before it becomes dynamic.

- Private VLAN ports cannot be dynamic-access ports.
- Dynamic-access ports cannot be members of an EtherChannel group.
- Port channels cannot be configured as dynamic-access ports.
- A dynamic-access port can participate in fallback bridging.
- The VTP management domain of the VMPS client and the VMPS server must be the same.
- The VLAN configured on the VMPS server should not be a voice VLAN.

Information About VMPS

Dynamic VLAN Assignments

The VLAN Query Protocol (VQP) is used to support dynamic-access ports, which are not permanently assigned to a VLAN, but give VLAN assignments based on the MAC source addresses seen on the port. Each time an unknown MAC address is seen, the device sends a VQP query to a remote VLAN Membership Policy Server (VMPS); the query includes the newly seen MAC address and the port on which it was seen. The VMPS responds with a VLAN assignment for the port. The device cannot be a VMPS server but can act as a client to the VMPS and communicate with it through VQP.

Each time the client device receives the MAC address of a new host, it sends a VQP query to the VMPS. When the VMPS receives this query, it searches its database for a MAC-address-to-VLAN mapping. The server response is based on this mapping and whether or not the server is in open or secure mode. In secure mode, the server shuts down the port when an illegal host is detected. In open mode, the server denies the host access to the port.

If the port is currently unassigned (that is, it does not yet have a VLAN assignment), the VMPS provides one of these responses:

- If the host is allowed on the port, the VMPS sends the client a vlan-assignment response containing the assigned VLAN name and allowing access to the host.
- If the host is not allowed on the port and the VMPS is in open mode, the VMPS sends an access-denied response.
- If the VLAN is not allowed on the port and the VMPS is in secure mode, the VMPS sends a port-shutdown response.

If the port already has a VLAN assignment, the VMPS provides one of these responses:

- If the VLAN in the database matches the current VLAN on the port, the VMPS sends a success response, allowing access to the host.
- If the VLAN in the database does not match the current VLAN on the port and active hosts exist on the port, the VMPS sends an access-denied or a port-shutdown response, depending on the secure mode of the VMPS.

If the device receives an access-denied response from the VMPS, it continues to block traffic to and from the host MAC address. The device continues to monitor the packets directed to the port and sends a query to the

VMPS when it identifies a new host address. If the device receives a port-shutdown response from the VMPS, it disables the port. The port must be manually reenabled by using Network Assistant, the CLI, or SNMP.

Related Topics

[Configuring Dynamic-Access Ports on VMPS Clients](#), on page 1765

[Example: VMPS Configuration](#), on page 1771

Dynamic-Access Port VLAN Membership

A dynamic-access port can belong to only one VLAN with an ID from 1 to 4094. When the link comes up, the device does not forward traffic to or from this port until the VMPS provides the VLAN assignment. The VMPS receives the source MAC address from the first packet of a new host connected to the dynamic-access port and attempts to match the MAC address to a VLAN in the VMPS database.

If there is a match, the VMPS sends the VLAN number for that port. If the client device was not previously configured, it uses the domain name from the first VTP packet it receives on its trunk port from the VMPS. If the client device was previously configured, it includes its domain name in the query packet to the VMPS to obtain its VLAN number. The VMPS verifies that the domain name in the packet matches its own domain name before accepting the request and responds to the client with the assigned VLAN number for the client. If there is no match, the VMPS either denies the request or shuts down the port (depending on the VMPS secure mode setting).

Multiple hosts (MAC addresses) can be active on a dynamic-access port if they are all in the same VLAN; however, the VMPS shuts down a dynamic-access port if more than 20 hosts are active on the port.

If the link goes down on a dynamic-access port, the port returns to an isolated state and does not belong to a VLAN. Any hosts that come online through the port are checked again through the VQP with the VMPS before the port is assigned to a VLAN.

Dynamic-access ports can be used for direct host connections, or they can connect to a network. A maximum of 20 MAC addresses are allowed per port on the device. A dynamic-access port can belong to only one VLAN at a time, but the VLAN can change over time, depending on the MAC addresses seen.

Related Topics

[Configuring Dynamic-Access Ports on VMPS Clients](#), on page 1765

[Example: VMPS Configuration](#), on page 1771

Default VMPS Client Configuration

The following table shows the default VMPS and dynamic-access port configuration on client switches.

Table 173: Default VMPS Client and Dynamic-Access Port Configuration

Feature	Default Setting
VMPS domain server	None
VMPS reconfirm interval	60 minutes
VMPS server retry count	3
Dynamic-access ports	None configured

How to Configure VMPS

Entering the IP Address of the VMPS



Note If the VMPS is being defined for a cluster of switches, enter the address on the command switch.

Before you begin

You must first enter the IP address of the server to configure the switch as a client.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **vmpls server *ipaddress* primary**
4. **vmpls server *ipaddress***
5. **end**
6. **show vmpls**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	vmpls server <i>ipaddress</i> primary Example: Device(config)# vmpls server 10.1.2.3 primary	Enters the IP address of the device acting as the primary VMPS server.
Step 4	vmpls server <i>ipaddress</i> Example:	(Optional) Enters the IP address of the device acting as a secondary VMPS server. You can enter up to three secondary server addresses.

	Command or Action	Purpose
	Device(config)# vmpls server 10.3.4.5	
Step 5	end Example: Device(config)# end	Returns to privileged EXEC mode.
Step 6	show vmpls Example: Device# show vmpls	Verifies your entries in the <i>VMPS Domain Server</i> field of the display.
Step 7	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Configuring Dynamic-Access Ports on VMPS Clients



Caution Dynamic-access port VLAN membership is for end stations or hubs connected to end stations. Connecting dynamic-access ports to other switches can cause a loss of connectivity.

If you are configuring a port on a cluster member device as a dynamic-access port, first use the **recommand** privileged EXEC command to log in to the cluster member device.

Before you begin

You must have IP connectivity to the VMPS for dynamic-access ports to work. You can test for IP connectivity by pinging the IP address of the VMPS and verifying that you get a response.



Note To return an interface to its default configuration, use the **default interface** *interface-id* interface configuration command. To return an interface to its default switchport mode (dynamic auto), use the **no switchport mode** interface configuration command. To reset the access mode to the default VLAN for the device, use the **no switchport access vlan** interface configuration command.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*

4. `switchport mode access`
5. `switchport access vlan dynamic`
6. `end`
7. `show interfaces interface-id switchport`
8. `copy running-config startup-config`

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> <code>enable</code>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode.
Step 3	interface interface-id Example: Device(config)# <code>interface gigabitethernet 0/1</code>	Specifies the device port that is connected to the end station, and enters interface configuration mode.
Step 4	switchport mode access Example: Device(config-if)# <code>switchport mode access</code>	Sets the port to access mode.
Step 5	switchport access vlan dynamic Example: Device(config-if)# <code>switchport access vlan dynamic</code>	Configures the port as eligible for dynamic VLAN membership. The dynamic-access port must be connected to an end station.
Step 6	end Example: Device(config)# <code>end</code>	Returns to privileged EXEC mode.
Step 7	show interfaces interface-id switchport Example: Device# <code>show interfaces gigabitethernet 0/1</code>	Verifies your entries in the <i>Operational Mode</i> field of the display.

	Command or Action	Purpose
	<code>switchport</code>	
Step 8	copy running-config startup-config Example: Device# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Related Topics

[Dynamic VLAN Assignments](#), on page 1762

[Example: VMPS Configuration](#), on page 1771

[Dynamic-Access Port VLAN Membership](#), on page 1763

Reconfirming VLAN Memberships

This task confirms the dynamic-access port VLAN membership assignments that the device has received from the VMPS.

SUMMARY STEPS

1. `enable`
2. `vmpls reconfirm`
3. `show vmpls`

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> <code>enable</code>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	vmpls reconfirm Example: Device# <code>vmpls reconfirm</code>	Reconfirms dynamic-access port VLAN membership.
Step 3	show vmpls Example: Device# <code>show vmpls</code>	Verifies the dynamic VLAN reconfirmation status.

Changing the Reconfirmation Interval

VMPS clients periodically reconfirm the VLAN membership information received from the VMPS. You can set the number of minutes after which reconfirmation occurs.



Note If you are configuring a member device in a cluster, this parameter must be equal to or greater than the reconfirmation setting on the command device. You also must first use the **command** privileged EXEC command to log in to the member device.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **vmpls reconfirm** *minutes*
4. **end**
5. **show vmpls**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	vmpls reconfirm <i>minutes</i> Example: Device(config)# vmpls reconfirm 90	Sets the number of minutes between reconfirmations of the dynamic VLAN membership. The range is 1 to 120. The default is 60 minutes. Note To return the switch to its default setting, use the no vmpls reconfirm global configuration command.
Step 4	end Example: Device(config)# end	Returns to privileged EXEC mode.

	Command or Action	Purpose
Step 5	show vmps Example: Device# <code>show vmps</code>	Verifies the dynamic VLAN reconfirmation status in the <i>Reconfirm Interval</i> field of the display.
Step 6	copy running-config startup-config Example: Device# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Changing the Retry Count

Follow these steps to change the number of times that the device attempts to contact the VMPS before querying the next server.

SUMMARY STEPS

1. `enable`
2. `configure terminal`
3. `vmps retry count`
4. `end`
5. `show vmps`
6. `copy running-config startup-config`

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> <code>enable</code>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode.
Step 3	vmps retry count Example: Device(config)# <code>vmps retry 5</code>	Changes the retry count. The retry range is 1 to 10; the default is 3. Note To return the device to its default setting, use the no vmps retry global configuration command.

	Command or Action	Purpose
Step 4	end Example: Device(config)# end	Returns to privileged EXEC mode.
Step 5	show vmmps Example: Device# show vmmps	Verifies your entry in the <i>Server Retry Count</i> field of the display.
Step 6	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Troubleshooting Dynamic-Access Port VLAN Membership

Problem The VMPS shuts down a dynamic-access port under these conditions:

- **Problem** The VMPS is in secure mode, and it does not allow the host to connect to the port. The VMPS shuts down the port to prevent the host from connecting to the network.
- **Problem** More than 20 active hosts reside on a dynamic-access port.

Solution To reenab a disabled dynamic-access port, enter the **shutdown** interface configuration command followed by the **no shutdown** interface configuration command.

Monitoring the VMPS

You can display information about the VMPS by using the **show vmmps** privileged EXEC command. The device displays this information about the VMPS:

- **VMPS VQP Version**—The version of VQP used to communicate with the VMPS. The device queries the VMPS that is using VQP Version 1.
- **Reconfirm Interval**—The number of minutes the device waits before reconfirming the VLAN-to-MAC-address assignments.
- **Server Retry Count**—The number of times VQP resends a query to the VMPS. If no response is received after this many tries, the device starts to query the secondary VMPS.
- **VMPS domain server**—The IP address of the configured VLAN membership policy servers. The device sends queries to the one marked *current*. The one marked *primary* is the primary server.

- **VMPS Action**—The result of the most recent reconfirmation attempt. A reconfirmation attempt can occur automatically when the reconfirmation interval expires, or you can force it by entering the **vmmps reconfirm** privileged EXEC command or its Network Assistant or SNMP equivalent.

This is an example of output for the **show vmmps** privileged EXEC command:

```
Device# show vmmps
VQP Client Status:
-----
VMPS VQP Version: 1
Reconfirm Interval: 60 min
Server Retry Count: 3
VMPS domain server: 172.20.128.86 (primary, current)
                   172.20.128.87

Reconfirmation status
-----
VMPS Action:          other
```

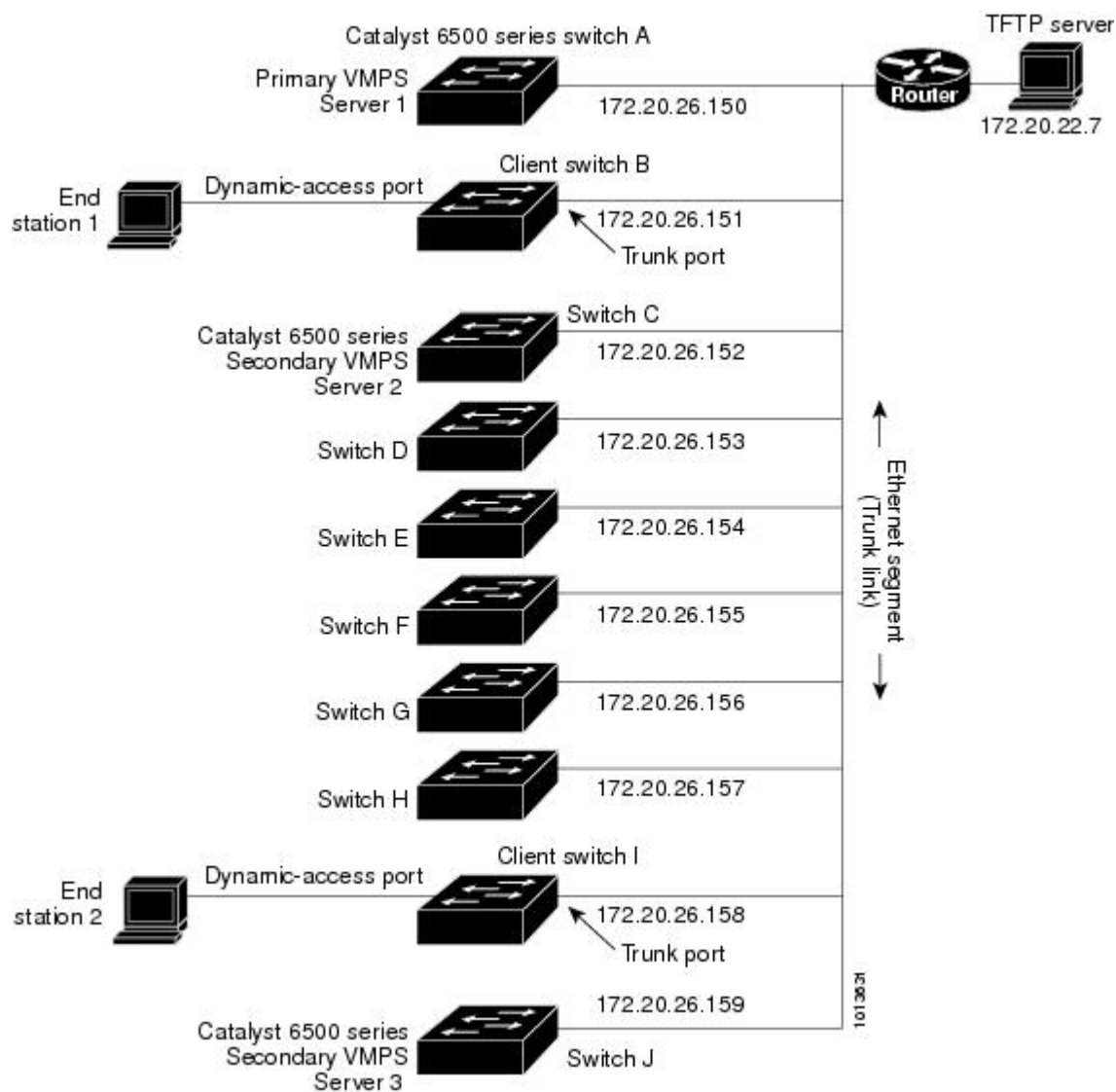
Configuration Example for VMPS

Example: VMPS Configuration

Figure 130: Dynamic Port VLAN Membership Configuration

This network has a VMPS server switch and VMPS client switches with dynamic-access ports with this configuration:

- The VMPS server and the VMPS client are separate switches.
- The Catalyst 6500 series Switch A is the primary VMPS server.
- The Catalyst 6500 series Switch C and Switch J are secondary VMPS servers.
- End stations are connected to the clients, Switch B and Switch I.
- The database configuration file is stored on the TFTP server with the IP address 172.20.22.7.



Related Topics

[Configuring Dynamic-Access Ports on VMPS Clients](#), on page 1765

[Dynamic VLAN Assignments](#), on page 1762

[Dynamic-Access Port VLAN Membership](#), on page 1763

Where to Go Next

You can configure the following:

- VTP
- VLANs
- VLAN Trunking

- Private VLANs
- Tunneling
- Voice VLANs

Additional References

Related Documents

Related Topic	Document Title
For complete syntax and usage information for the commands used in this chapter.	<i>Catalyst 2960-XR Switch VLAN Management Command Reference</i>

Standards and RFCs

Standard/RFC	Title
—	—

MIBs

MIB	MIBs Link
All the supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
<p>The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies.</p> <p>To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds.</p> <p>Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.</p>	http://www.cisco.com/support

Feature History and Information for VMPS

Release	Modification
Cisco IOS Release 15.0(2)EX1	This feature was introduced.



CHAPTER 89

Configuring IEEE 802.1Q and Layer 2 Protocol Tunneling

- [Prerequisites for Configuring Tunneling, on page 1775](#)
- [Information about Tunneling, on page 1777](#)
- [How to Configure Tunneling, on page 1785](#)
- [Configuration Examples for IEEE 802.1Q and Layer 2 Protocol Tunneling, on page 1796](#)
- [Monitoring Tunneling Status, on page 1798](#)
- [Where to Go Next, on page 1799](#)
- [Additional References, on page 1799](#)
- [Feature History and Information for Tunneling, on page 1800](#)

Prerequisites for Configuring Tunneling

The following sections list prerequisites and considerations for configuring IEEE 802.1Q and Layer 2 protocol tunneling.

IEEE 802.1Q Tunneling

Although IEEE 802.1Q tunneling works well for Layer 2 packet switching, there are incompatibilities between some Layer 2 features and Layer 3 switching.

- A tunnel port cannot be a routed port.
- IP routing is not supported on a VLAN that includes IEEE 802.1Q tunnel ports. Packets received from a tunnel port are forwarded based only on Layer 2 information. If routing is enabled on a device virtual interface (SVI) that includes tunnel ports, untagged IP packets received from the tunnel port are recognized and routed by the device. Customers can access the Internet through its native VLAN. If this access is not needed, you should not configure SVIs on VLANs that include tunnel ports.
- Fallback bridging is not supported on tunnel ports. Because all IEEE 802.1Q-tagged packets received from a tunnel port are treated as non-IP packets, if fallback bridging is enabled on VLANs that have tunnel ports configured, IP packets would be improperly bridged across VLANs. Therefore, you must not enable fallback bridging on VLANs with tunnel ports.
- Tunnel ports do not support IP access control lists (ACLs).

- Layer 3 quality of service (QoS) ACLs and other QoS features related to Layer 3 information are not supported on tunnel ports. MAC-based QoS is supported on tunnel ports.
- EtherChannel port groups are compatible with tunnel ports as long as the IEEE 802.1Q configuration is consistent within an EtherChannel port group.
- Port Aggregation Protocol (PAgP), Link Aggregation Control Protocol (LACP), and UniDirectional Link Detection (UDLD) are supported on IEEE 802.1Q tunnel ports.
- Dynamic Trunking Protocol (DTP) is not compatible with IEEE 802.1Q tunneling because you must manually configure asymmetric links with tunnel ports and trunk ports.
- VLAN Trunking Protocol (VTP) does not work between devices that are connected by an asymmetrical link or devices that communicate through a tunnel.
- Loopback detection is supported on IEEE 802.1Q tunnel ports.
- When a port is configured as an IEEE 802.1Q tunnel port, spanning-tree bridge protocol data unit (BPDU) filtering is automatically enabled on the interface. Cisco Discovery Protocol (CDP) and the Layer Link Discovery Protocol (LLDP) are automatically disabled on the interface.

Layer 2 Protocol Tunneling

- The device supports tunneling of CDP, STP, including multiple STP (MSTP), and VTP. Protocol tunneling is disabled by default but can be enabled for the individual protocols on IEEE 802.1Q tunnel ports or access ports.
- The device does not support Layer 2 protocol tunneling on ports with switchport mode dynamic auto or dynamic desirable.
- DTP is not compatible with layer 2 protocol tunneling.
- The edge devices on the outbound side of the service-provider network restore the proper Layer 2 protocol and MAC address information and forward the packets to all tunnel and access ports in the same metro VLAN.
- For interoperability with third-party vendor devices, the device supports a Layer 2 protocol-tunnel bypass feature. Bypass mode transparently forwards control PDUs to vendor devices that have different ways of controlling protocol tunneling. When Layer 2 protocol tunneling is enabled on ingress ports on a device, egress trunk ports forward the tunneled packets with a special encapsulation. If you also enable Layer 2 protocol tunneling on the egress trunk port, this behavior is bypassed, and the device forwards control PDUs without any processing or modification.
- The device supports PAgP, LACP, and UDLD tunneling for emulated point-to-point network topologies. Protocol tunneling is disabled by default but can be enabled for the individual protocols on IEEE 802.1Q tunnel ports or on access ports.
- If you enable PAgP or LACP tunneling, we recommend that you also enable UDLD on the interface for faster link-failure detection.
- Loopback detection is not supported on Layer 2 protocol tunneling of PAgP, LACP, or UDLD packets.
- EtherChannel port groups are compatible with tunnel ports when the IEEE 802.1Q configuration is consistent within an EtherChannel port group.

- If an encapsulated PDU (with the proprietary destination MAC address) is received from a tunnel port or an access port with Layer 2 tunneling enabled, the tunnel port is shut down to prevent loops. The port also shuts down when a configured shutdown threshold for the protocol is reached. You can manually reenab the port (by entering a **shutdown** and a **no shutdown** command sequence). If errdisable recovery is enabled, the operation is retried after a specified time interval.
- Only decapsulated PDUs are forwarded to the customer network. The spanning-tree instance running on the service-provider network does not forward BPDUs to tunnel ports. CDP packets are not forwarded from tunnel ports.
- When protocol tunneling is enabled on an interface, you can set a per-protocol, per-port, shutdown threshold for the PDUs generated by the customer network. If the limit is exceeded, the port shuts down. You can also limit BPDU rate by using QoS ACLs and policy maps on a tunnel port.
- When protocol tunneling is enabled on an interface, you can set a per-protocol, per-port, drop threshold for the PDUs generated by the customer network. If the limit is exceeded, the port drops PDUs until the rate at which it receives them is below the drop threshold.
- Because tunneled PDUs (especially STP BPDUs) must be delivered to all remote sites so that the customer virtual network operates properly, you can give PDUs higher priority within the service-provider network than data packets received from the same tunnel port. By default, the PDUs use the same CoS value as data packets.

Layer 2 Tunneling for EtherChannels

To configure Layer 2 point-to-point tunneling to facilitate the automatic creation of EtherChannels, you need to configure both the SP (service-provider) edge switch and the customer device.

Information about Tunneling

IEEE 802.1Q and Layer 2 Protocol Overview

Virtual private networks (VPNs) provide enterprise-scale connectivity on a shared infrastructure, often Ethernet-based, with the same security, prioritization, reliability, and manageability requirements of private networks. Tunneling is a feature designed for service providers who carry traffic of multiple customers across their networks and are required to maintain the VLAN and Layer 2 protocol configurations of each customer without impacting the traffic of other customers.

For complete syntax and usage information for the commands used in this chapter, see the command reference for this release.

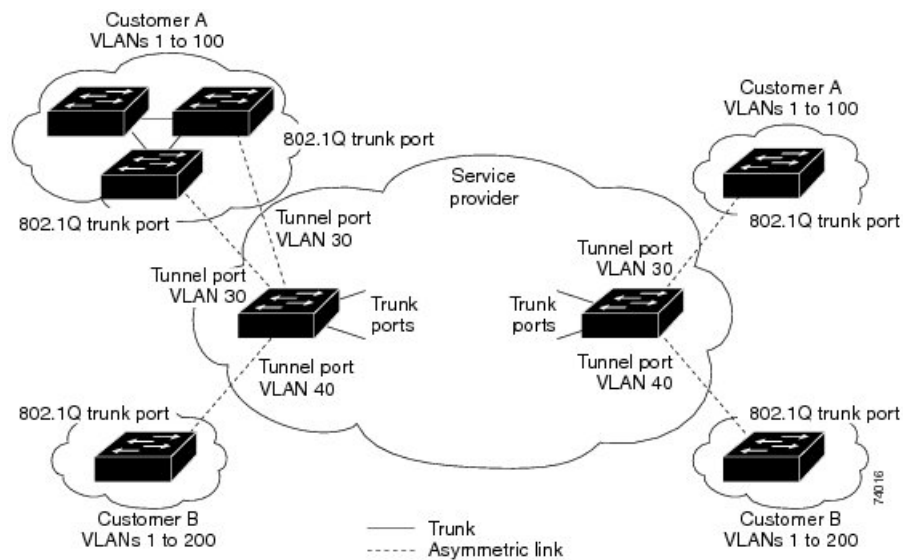
IEEE 802.1Q Tunneling

Business customers of service providers often have specific requirements for VLAN IDs and the number of VLANs to be supported. The VLAN ranges required by different customers in the same service-provider network might overlap, and traffic of customers through the infrastructure might be mixed. Assigning a unique range of VLAN IDs to each customer would restrict customer configurations and could easily exceed the VLAN limit (4096) of the IEEE 802.1Q specification.

Using the IEEE 802.1Q tunneling feature, service providers can use a single VLAN to support customers who have multiple VLANs. Customer VLAN IDs are preserved, and traffic from different customers is segregated within the service-provider network, even when they appear to be in the same VLAN. Using IEEE 802.1Q tunneling expands VLAN space by using a VLAN-in-VLAN hierarchy and retagging the tagged packets. A port configured to support IEEE 802.1Q tunneling is called a tunnel port. When you configure tunneling, you assign a tunnel port to a VLAN ID that is dedicated to tunneling. Each customer requires a separate service-provider VLAN ID, but that VLAN ID supports all of the customer's VLANs.

Customer traffic tagged in the normal way with appropriate VLAN IDs comes from an IEEE 802.1Q trunk port on the customer device and into a tunnel port on the service-provider edge device. The link between the customer device and the edge device is asymmetric because one end is configured as an IEEE 802.1Q trunk port, and the other end is configured as a tunnel port. You assign the tunnel port interface to an access VLAN ID that is unique to each customer.

Figure 131: IEEE 802.1Q Tunnel Ports in a Service-Provider Network

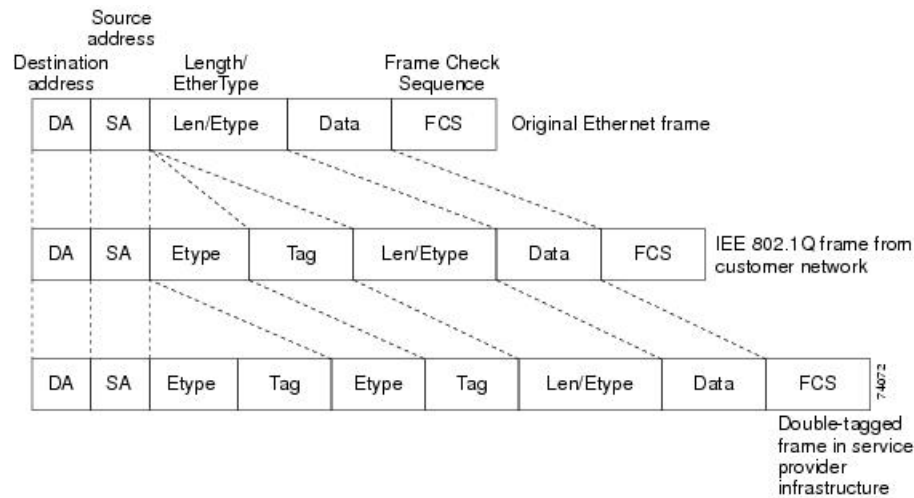


Packets coming from the customer trunk port into the tunnel port on the service-provider edge device are normally IEEE 802.1Q-tagged with the appropriate VLAN ID. The tagged packets remain intact inside the device and when they exit the trunk port into the service-provider network, they are encapsulated with another layer of an IEEE 802.1Q tag (called the metro tag) that contains the VLAN ID that is unique to the customer. The original customer IEEE 802.1Q tag is preserved in the encapsulated packet. Therefore, packets entering the service-provider network are double-tagged, with the outer (metro) tag containing the customer's access VLAN ID, and the inner VLAN ID being that of the incoming traffic.

When the double-tagged packet enters another trunk port in a service-provider core device, the outer tag is stripped as the device processes the packet. When the packet exits another trunk port on the same core device, the same metro tag is again added to the packet.

Figure 132: Original (Normal), IEEE 802.1Q, and Double-Tagged Ethernet Packet Formats

This figure shows the tag structures of the double-tagged packets.



When the packet enters the trunk port of the service-provider egress device, the outer tag is again stripped as the device internally processes the packet. However, the metro tag is not added when the packet is sent out the tunnel port on the edge device into the customer network. The packet is sent as a normal IEEE 802.1Q-tagged frame to preserve the original VLAN numbers in the customer network.

In the above network figure, Customer A was assigned VLAN 30, and Customer B was assigned VLAN 40. Packets entering the edge device tunnel ports with IEEE 802.1Q tags are double-tagged when they enter the service-provider network, with the outer tag containing VLAN ID 30 or 40, appropriately, and the inner tag containing the original VLAN number, for example, VLAN 100. Even if both Customers A and B have VLAN 100 in their networks, the traffic remains segregated within the service-provider network because the outer tag is different. Each customer controls its own VLAN numbering space, which is independent of the VLAN numbering space used by other customers and the VLAN numbering space used by the service-provider network.

At the outbound tunnel port, the original VLAN numbers on the customer's network are recovered. It is possible to have multiple levels of tunneling and tagging, but the device supports only one level in this release.

If traffic coming from a customer network is not tagged (native VLAN frames), these packets are bridged or routed as normal packets. All packets entering the service-provider network through a tunnel port on an edge device are treated as untagged packets, whether they are untagged or already tagged with IEEE 802.1Q headers. The packets are encapsulated with the metro tag VLAN ID (set to the access VLAN of the tunnel port) when they are sent through the service-provider network on an IEEE 802.1Q trunk port. The priority field on the metro tag is set to the interface class of service (CoS) priority configured on the tunnel port. (The default is zero if none is configured.)

On device, because 802.1Q tunneling is configured on a per-port basis, it does not matter whether the device is a standalone device or a stack member. All configuration is done on the active stack.

IEEE 802.1Q Tunneling Configuration Guidelines

When you configure IEEE 802.1Q tunneling, you should always use an asymmetrical link between the customer device and the edge device, with the customer device port configured as an IEEE 802.1Q trunk port and the edge device port configured as a tunnel port.

Assign tunnel ports only to VLANs that are used for tunneling.

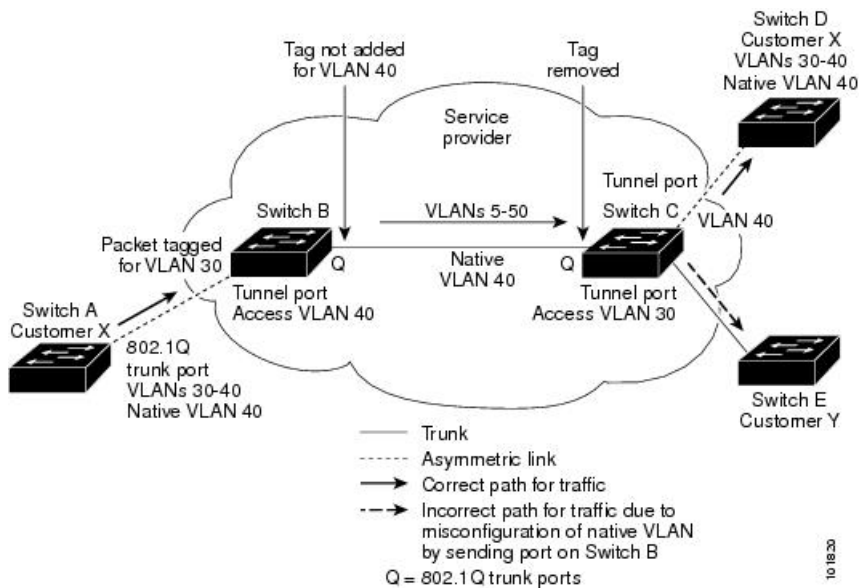
Configuration requirements for native VLANs and for and maximum transmission units (MTUs) are explained in these next sections.

Native VLANs

When configuring IEEE 802.1Q tunneling on an edge device, you must use IEEE 802.1Q trunk ports for sending packets into the service-provider network. However, packets going through the core of the service-provider network can be carried through IEEE 802.1Q trunks, ISL trunks, or nontrunking links. When IEEE 802.1Q trunks are used in these core devices, the native VLANs of the IEEE 802.1Q trunks must not match any native VLAN of the nontrunking (tunneling) port on the same device because traffic on the native VLAN would not be tagged on the IEEE 802.1Q sending trunk port.

In the following network figure, VLAN 40 is configured as the native VLAN for the IEEE 802.1Q trunk port from Customer X at the ingress edge device in the service-provider network (Device B). Device A of Customer X sends a tagged packet on VLAN 30 to the ingress tunnel port of Device B in the service-provider network, which belongs to access VLAN 40. Because the access VLAN of the tunnel port (VLAN 40) is the same as the native VLAN of the edge device trunk port (VLAN 40), the metro tag is not added to tagged packets received from the tunnel port. The packet carries only the VLAN 30 tag through the service-provider network to the trunk port of the egress-edgedevice (Device C) and is misdirected through the egress device tunnel port to Customer Y.

Figure 133: Potential Problems with IEEE 802.1Q Tunneling and Native VLANs



These are some ways to solve this problem:

- Use the **vlan dot1q tag native** global configuration command to configure the edge devices so that all packets going out an IEEE 802.1Q trunk, including the native VLAN, are tagged. If the device is configured to tag native VLAN packets on all IEEE 802.1Q trunks, the device accepts untagged packets, but sends only tagged packets.

- Ensure that the native VLAN ID on the edge devices trunk port is not within the customer VLAN range. For example, if the trunk port carries traffic of VLANs 100 to 200, assign the native VLAN a number outside that range.

System MTU

The default system MTU for traffic on the device is 1500 bytes. You can configure Fast Ethernet ports on the device members in the mixed hardware device stack to support frames larger than 1500 bytes by using the **system mtu** global configuration command.

You can configure 10-Gigabit and Gigabit Ethernet ports to support frames larger than 1500 bytes by using the **system mtu jumbo** global configuration command.

The system MTU and system jumbo MTU values do not include the IEEE 802.1Q header. Because the IEEE 802.1Q tunneling feature increases the frame size by 4 bytes when the metro tag is added, you must configure all devices in the service-provider network to be able to process maximum frames by adding 4 bytes to the system MTU and system jumbo MTU sizes.

For example, the device supports a maximum frame size of 1496 bytes with one of these configurations:

- The device has a system jumbo MTU value of 1500 bytes, and the **switchport mode dot1q tunnel** interface configuration command is configured on a 10-Gigabit or Gigabit Ethernet device port.
- The device member has a system MTU value of 1500 bytes, and the **switchport mode dot1q tunnel** interface configuration command is configured on a Fast Ethernet port of the member.

Default IEEE 802.1Q Tunneling Configuration

By default, IEEE 802.1Q tunneling is disabled because the default switchport mode is dynamic auto. Tagging of IEEE 802.1Q native VLAN packets on all IEEE 802.1Q trunk ports is also disabled.

Layer 2 Protocol Tunneling Overview

Customers at different sites connected across a service-provider network need to use various Layer 2 protocols to scale their topologies to include all remote sites, as well as the local sites. STP must run properly, and every VLAN should build a proper spanning tree that includes the local site and all remote sites across the service-provider network. Cisco Discovery Protocol (CDP) must discover neighboring Cisco devices from local and remote sites. VLAN Trunking Protocol (VTP) must provide consistent VLAN configuration throughout all sites in the customer network.

When protocol tunneling is enabled, edge devices on the inbound side of the service-provider network encapsulate Layer 2 protocol packets with a special MAC address and send them across the service-provider network. Core devices in the network do not process these packets but forward them as normal packets. Layer 2 protocol data units (PDUs) for CDP, STP, or VTP cross the service-provider network and are delivered to customer devices on the outbound side of the service-provider network. Identical packets are received by all customer ports on the same VLANs with these results:

- Users on each of a customer's sites can properly run STP, and every VLAN can build a correct spanning tree based on parameters from all sites and not just from the local site.
- CDP discovers and shows information about the other Cisco devices connected through the service-provider network.

- VTP provides consistent VLAN configuration throughout the customer network, propagating to all devices through the service provider.



Note To provide interoperability with third-party vendors, you can use the Layer 2 protocol-tunnel bypass feature. Bypass mode transparently forwards control PDUs to vendor devices that have different ways of controlling protocol tunneling. You implement bypass mode by enabling Layer 2 protocol tunneling on the egress trunk port. When Layer 2 protocol tunneling is enabled on the trunk port, the encapsulated tunnel MAC address is removed and the protocol packets have their normal MAC address.

Layer 2 protocol tunneling can be used independently or can enhance IEEE 802.1Q tunneling. If protocol tunneling is not enabled on IEEE 802.1Q tunneling ports, remote devices at the receiving end of the service-provider network do not receive the PDUs and cannot properly run STP, CDP, and VTP. When protocol tunneling is enabled, Layer 2 protocols within each customer's network are totally separate from those running within the service-provider network. Customer devices on different sites that send traffic through the service-provider network with IEEE 802.1Q tunneling achieve complete knowledge of the customer's VLAN. If IEEE 802.1Q tunneling is not used, you can still enable Layer 2 protocol tunneling by connecting to the customer device through access ports and by enabling tunneling on the service-provider access port.

For example, in the following figure (Layer 2 Protocol Tunneling), Customer X has four devices in the same VLAN, that are connected through the service-provider network. If the network does not tunnel PDUs, devices on the far ends of the network cannot properly run STP, CDP, and VTP. For example, STP for a VLAN on a device in Customer X, Site 1, will build a spanning tree on the devices at that site without considering convergence parameters based on Customer X's device in Site 2. This could result in the topology shown in the Layer 2 Network Topology without Proper Convergence figure.

Figure 134: Layer 2 Protocol Tunneling

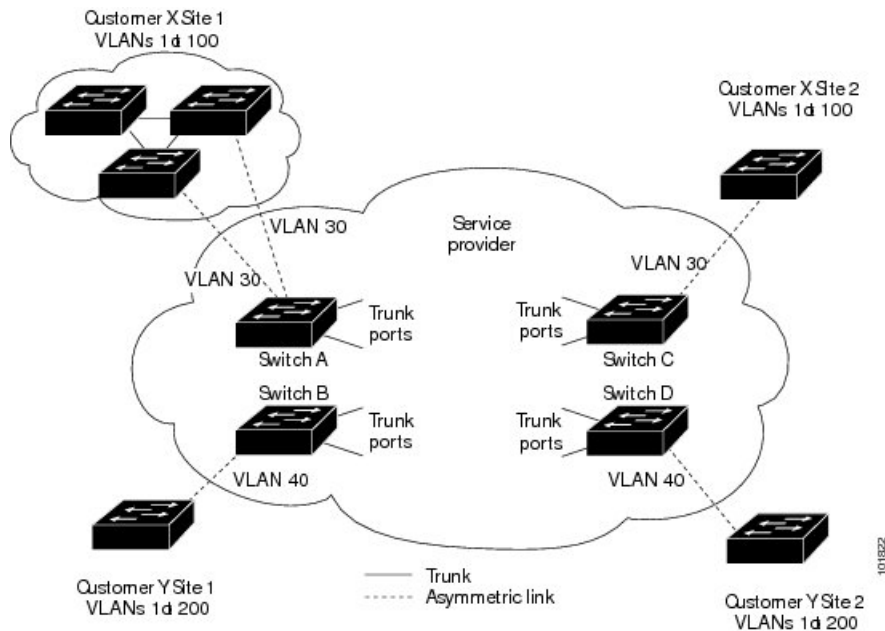
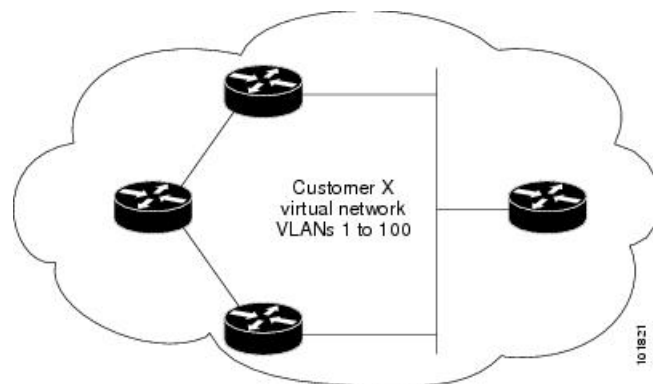


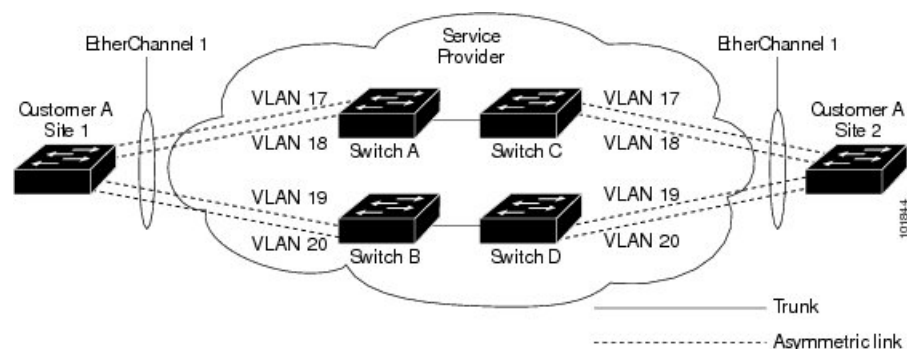
Figure 135: Layer 2 Network Topology Without Proper Convergence



In an SP network, you can use Layer 2 protocol tunneling to enhance the creation of EtherChannels by emulating a point-to-point network topology. When you enable protocol tunneling (PAGP or LACP) on the SP device, remote customer devices receive the PDUs and can negotiate the automatic creation of EtherChannels.

For example, in the following figure (Layer 2 Protocol Tunneling for EtherChannels), Customer A has two devices in the same VLAN that are connected through the SP network. When the network tunnels PDUs, devices on the far ends of the network can negotiate the automatic creation of EtherChannels without needing dedicated lines.

Figure 136: Layer 2 Protocol Tunneling for EtherChannels



Layer 2 Protocol Tunneling on Ports

You can enable Layer 2 protocol tunneling (by protocol) on the ports that are connected to the customer in the edge devices of the service-provider network. The service-provider edge devices connected to the customer device perform the tunneling process. Edge device tunnel ports are connected to customer IEEE 802.1Q trunk ports. Edge device access ports are connected to customer access ports. The edge devices connected to the customer device perform the tunneling process.

You can enable Layer 2 protocol tunneling on ports that are configured as access ports or tunnel ports. You cannot enable Layer 2 protocol tunneling on ports configured in either **switchport mode dynamic auto** mode (the default mode) or **switchport mode dynamic desirable** mode.

The device supports Layer 2 protocol tunneling for CDP, STP, and VTP. For emulated point-to-point network topologies, it also supports PAGP, LACP, and UDLD protocols. The device does not support Layer 2 protocol tunneling for LLDP.



Note PAgP, LACP, and UDLD protocol tunneling is only intended to emulate a point-to-point topology. An erroneous configuration that sends tunneled packets to many ports could lead to a network failure.

When the Layer 2 PDUs that entered the service-provider inbound edge device through a Layer 2 protocol-enabled port exit through the trunk port into the service-provider network, the device overwrites the customer PDU-destination MAC address with a well-known Cisco proprietary multicast address (01-00-0c-cd-cd-d0). If IEEE 802.1Q tunneling is enabled, packets are also double-tagged; the outer tag is the customer metro tag, and the inner tag is the customer's VLAN tag. The core devices ignore the inner tags and forward the packet to all trunk ports in the same metro VLAN. The edge devices on the outbound side restore the proper Layer 2 protocol and MAC address information and forward the packets to all tunnel or access ports in the same metro VLAN. Therefore, the Layer 2 PDUs remain intact and are delivered across the service-provider infrastructure to the other side of the customer network.

See the Layer 2 Protocol Tunneling figure in [Layer 2 Protocol Tunneling Overview, on page 1781](#), with Customer X and Customer Y in access VLANs 30 and 40, respectively. Asymmetric links connect the customers in Site 1 to edge devices in the service-provider network. The Layer 2 PDUs (for example, BPDUs) coming into Device B from Customer Y in Site 1 are forwarded to the infrastructure as double-tagged packets with the well-known MAC address as the destination MAC address. These double-tagged packets have the metro VLAN tag of 40, as well as an inner VLAN tag (for example, VLAN 100). When the double-tagged packets enter Device D, the outer VLAN tag 40 is removed, the well-known MAC address is replaced with the respective Layer 2 protocol MAC address, and the packet is sent to Customer Y on Site 2 as a single-tagged frame in VLAN 100.

You can also enable Layer 2 protocol tunneling on access ports on the edge device connected to access or trunk ports on the customer device. In this case, the encapsulation and decapsulation process is the same as described in the previous paragraph, except that the packets are not double-tagged in the service-provider network. The single tag is the customer-specific access VLAN tag.

In device stacks, Layer 2 protocol tunneling configuration is distributed among all stack members. Each stack member that receives an ingress packet on a local port encapsulates or decapsulates the packet and forwards it to the appropriate destination port. On a single device, ingress Layer 2 protocol-tunneled traffic is sent across all local ports in the same VLAN on which Layer 2 protocol tunneling is enabled. In a stack, packets received by a Layer 2 protocol-tunneled port are distributed to all ports in the stack that are configured for Layer 2 protocol tunneling and are in the same VLAN. All Layer 2 protocol tunneling configuration is handled by the active stack and distributed to all stack members.

Default Layer 2 Protocol Tunneling Configuration

The following table shows the default Layer 2 protocol tunneling configuration.

Table 174: Default Layer 2 Ethernet Interface VLAN Configuration

Feature	Default Setting
Layer 2 protocol tunneling	Disabled.
Shutdown threshold	None set.
Drop threshold	None set.

Feature	Default Setting
CoS Value	If a CoS value is configured on the interface, that value is used to set the BPDU CoS value for Layer 2 protocol tunneling. If no CoS value is configured at the interface level, the default value for CoS marking of L2 protocol tunneling BPDUs is 5. This does not apply to data traffic.

How to Configure Tunneling

Configuring an IEEE 802.1Q Tunneling Port

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface *interface-id***
4. **switchport access vlan *vlan-id***
5. **switchport mode dot1q-tunnel**
6. **exit**
7. **vlan dot1q tag native**
8. **end**
9. Use one of the following:
 - **show dot1q-tunnel**
 - **show running-config interface**
10. **show vlan dot1q tag native**
11. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.

	Command or Action	Purpose
Step 3	interface <i>interface-id</i> Example: Device (config) # interface gigabitethernet2/0/1	Enters interface configuration mode for the interface to be configured as a tunnel port. This should be the edge port in the service-provider network that connects to the customer device. Valid interfaces include physical interfaces and port-channel logical interfaces (port channels 1 to 48).
Step 4	switchport access vlan <i>vlan-id</i> Example: Device (config-if) # switchport access vlan 2	Specifies the default VLAN, which is used if the interface stops trunking. This VLAN ID is specific to the particular customer.
Step 5	switchport mode dot1q-tunnel Example: Device (config-if) # switchport mode dot1q-tunnel	Sets the interface as an IEEE 802.1Q tunnel port. Note Use the no switchport mode dot1q-tunnel interface configuration command to return the port to the default state of dynamic desirable.
Step 6	exit Example: Device (config-if) # exit	Returns to privileged EXEC mode.
Step 7	vlan dot1q tag native Example: Device (config) # vlan dot1q tag native	(Optional) Sets the device to enable tagging of native VLAN packets on all IEEE 802.1Q trunk ports. When not set, and a customer VLAN ID is the same as the native VLAN, the trunk port does not apply a metro tag, and packets could be sent to the wrong destination. Note Use the no vlan dot1q tag native global configuration command to disable tagging of native VLAN packets.
Step 8	end Example: Device (config) # end	Returns to privileged EXEC mode.
Step 9	Use one of the following: <ul style="list-style-type: none"> • show dot1q-tunnel • show running-config interface Example: Device# show dot1q-tunnel	Displays the ports configured for IEEE 802.1Q tunneling. Displays the ports that are in tunnel mode.

	Command or Action	Purpose
	OR Device# <code>show running-config interface</code>	
Step 10	show vlan dot1q tag native Example: Device# <code>show vlan dot1q native</code>	Displays IEEE 802.1Q native VLAN tagging status.
Step 11	copy running-config startup-config Example: Device# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Configuring Layer 2 Protocol Tunneling

SUMMARY STEPS

1. `enable`
2. `configure terminal`
3. `interface interface-id`
4. Use one of the following:
 - `switchport mode access`
 - `switchport mode dot1q-tunnel`
5. `l2protocol-tunnel [cdp | lldp | point-to-point | stp | vtp]`
6. `l2protocol-tunnel shutdown-threshold [packet_second_rate_value | cdp | lldp point-to-point | stp | vtp]`
7. `l2protocol-tunnel drop-threshold [packet_second_rate_value | cdp | lldp | point-to-point | stp | vtp]`
8. `exit`
9. `errdisable recovery cause l2ptguard`
10. `l2protocol-tunnel cos value`
11. `end`
12. `show l2protocol`
13. `copy running-config startup-config`

DETAILED STEPS

	Command or Action	Purpose
Step 1	<code>enable</code> Example:	Enables privileged EXEC mode. • Enter your password if prompted.

	Command or Action	Purpose
	Device> <code>enable</code>	
Step 2	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode.
Step 3	interface <i>interface-id</i> Example: Device(config)# <code>interface gigabitethernet1/0/1</code>	Specifies the interface connected to the phone, and enters interface configuration mode.
Step 4	Use one of the following: <ul style="list-style-type: none"> • <code>switchport mode access</code> • <code>switchport mode dot1q-tunnel</code> Example: Device# <code>switchport mode access</code> OR Device# <code>switchport mode dot1q-tunnel</code>	Configures the interface as an access port or an IEEE 802.1Q tunnel port.
Step 5	l2protocol-tunnel [cdp lldp point-to-point stp vtp] Example: Device# <code>l2protocol-tunnel cdp</code>	Enables protocol tunneling for the desired protocol. If no keyword is entered, tunneling is enabled for all three Layer 2 protocols. Note Use the <code>no l2protocol-tunnel [cdp lldp point-to-point stp vtp]</code> interface configuration command to disable protocol tunneling for one of the Layer 2 protocols or for all three.
Step 6	l2protocol-tunnel shutdown-threshold [<i>packet_second_rate_value</i> cdp lldp point-to-point stp vtp] Example: Device# <code>l2protocol-tunnel shutdown-threshold 100 cdp</code>	(Optional) Configures the threshold for packets-per-second accepted for encapsulation. The interface is disabled if the configured threshold is exceeded. If no protocol option is specified, the threshold applies to each of the tunneled Layer 2 protocol types. The range is 1 to 4096. The default is to have no threshold configured. Note If you also set a drop threshold on this interface, the <code>shutdown-threshold</code> value must be greater than or equal to the <code>drop-threshold</code> value.

	Command or Action	Purpose
		<p>Note Use the no l2protocol-tunnel shutdown-threshold [<i>packet_second_rate_value</i> cdp lldp point-to-point stp vtp] and the no l2protocol-tunnel drop-threshold [<i>packet_second_rate_value</i> cdp lldp point-to-point stp vtp] commands to return the shutdown and drop thresholds to the default settings.</p>
Step 7	<p>l2protocol-tunnel drop-threshold [<i>packet_second_rate_value</i> cdp lldp point-to-point stp vtp]</p> <p>Example:</p> <pre>Device# l2protocol-tunnel drop-threshold 100 cdp</pre>	<p>(Optional) Configures the threshold for packets-per-second accepted for encapsulation. The interface drops packets if the configured threshold is exceeded. If no protocol option is specified, the threshold applies to each of the tunneled Layer 2 protocol types. The range is 1 to 4096. The default is to have no threshold configured.</p> <p>Note If you also set a shutdown threshold on this interface, the drop-threshold value must be less than or equal to the shutdown-threshold value.</p> <p>Note Use the no l2protocol-tunnel shutdown-threshold [cdp lldp point-to-point stp vtp] and the no l2protocol-tunnel drop-threshold [cdp stp vtp] commands to return the shutdown and drop thresholds to the default settings.</p>
Step 8	<p>exit</p> <p>Example:</p> <pre>Device# exit</pre>	Returns to global configuration mode.
Step 9	<p>errdisable recovery cause l2ptguard</p> <p>Example:</p> <pre>Device(config)# errdisable recovery cause l2ptguard</pre>	(Optional) Configures the recovery mechanism from a Layer 2 maximum-rate error so that the interface is reenabled and can try again. Errdisable recovery is disabled by default; when enabled, the default time interval is 300 seconds.
Step 10	<p>l2protocol-tunnel cos value</p> <p>Example:</p> <pre>Device(config)# l2protocol-tunnel cos value 7</pre>	(Optional) Configures the CoS value for all tunneled Layer 2 PDUs. The range is 0 to 7; the default is the default CoS value for the interface. If none is configured, the default is 5.

	Command or Action	Purpose
Step 11	end Example: Device(config)# end	Returns to privileged EXEC mode.
Step 12	show l2protocol Example: Device# show l2protocol	Displays the Layer 2 tunnel ports on the device, including the protocols configured, the thresholds, and the counters.
Step 13	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Configuring the SP Edge Switch

Before you begin

For EtherChannels, you need to configure both the SP (service-provider) edge devices and the customer devices for Layer 2 protocol tunneling.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **switchport mode dot1q-tunnel**
5. **l2protocol-tunnel point-to-point** [**pagp** | **lACP** | **udld**]
6. **l2protocol-tunnel shutdown-threshold** [**point-to-point** [**pagp** | **lACP** | **udld**]] *value*
7. **l2protocol-tunnel drop-threshold** [**point-to-point** [**pagp** | **lACP** | **udld**]] *value*
8. **no cdp enable**
9. **spanning-tree bpdu filter enable**
10. **exit**
11. **errdisable recovery cause l2ptguard**
12. **l2protocol-tunnel cos** *value*
13. **end**
14. **show l2protocol**
15. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	interface interface-id Example: Device(config)# interface gigabitethernet1/0/1	Specifies the interface connected to the phone, and enters interface configuration mode.
Step 4	switchport mode dot1q-tunnel Example: Device(config-if)# switchport mode dot1q-tunnel	Configures the interface as an IEEE 802.1Q tunnel port.
Step 5	l2protocol-tunnel point-to-point [pagp lacp udld] Example: Device(config-if)# l2protocol-tunnel point-to-point pagp	(Optional) Enables point-to-point protocol tunneling for the desired protocol. If no keyword is entered, tunneling is enabled for all three protocols. <p>Note To avoid a network failure, make sure that the network is a point-to-point topology before you enable tunneling for PAGP, LACP, or UDLD packets.</p> <p>Note Use the no l2protocol-tunnel [point-to-point [pagp lacp udld]] interface configuration command to disable point-to-point protocol tunneling for one of the Layer 2 protocols or for all three.</p>
Step 6	l2protocol-tunnel shutdown-threshold [point-to-point [pagp lacp udld]] value Example: Device(config-if)# l2protocol-tunnel shutdown-threshold point-to-point pagp 100	(Optional) Configures the threshold for packets-per-second accepted for encapsulation. The interface is disabled if the configured threshold is exceeded. If no protocol option is specified, the threshold applies to each of the tunneled Layer 2 protocol types. The range is 1 to 4096. The default is to have no threshold configured.

	Command or Action	Purpose
		<p>Note If you also set a drop threshold on this interface, the shutdown-threshold value must be greater than or equal to the drop-threshold value.</p> <p>Note Use the no l2protocol-tunnel shutdown-threshold [point-to-point [pagp lacp udld]] and the no l2protocol-tunnel drop-threshold [[point-to-point [pagp lacp udld]] commands to return the shutdown and drop thresholds to the default settings.</p>
Step 7	<p>l2protocol-tunnel drop-threshold [point-to-point [pagp lacp udld]] value</p> <p>Example:</p> <pre>Device(config-if)# l2protocol-tunnel drop-threshold point-to-point pagp 500</pre>	<p>(Optional) Configures the threshold for packets-per-second accepted for encapsulation. The interface drops packets if the configured threshold is exceeded. If no protocol option is specified, the threshold applies to each of the tunneled Layer 2 protocol types. The range is 1 to 4096. The default is to have no threshold configured.</p> <p>Note If you also set a shutdown threshold on this interface, the drop-threshold value must be less than or equal to the shutdown-threshold value.</p>
Step 8	<p>no cdp enable</p> <p>Example:</p> <pre>Device(config-if)# no cdp enable</pre>	Disables CDP on the interface.
Step 9	<p>spanning-tree bpdu filter enable</p> <p>Example:</p> <pre>Device(config-if)# spanning-tree bpdu filter enable</pre>	Enables BPDU filtering on the interface.
Step 10	<p>exit</p> <p>Example:</p> <pre>Device(config-if)# exit</pre>	Returns to global configuration mode.
Step 11	<p>errdisable recovery cause l2ptguard</p> <p>Example:</p> <pre>Device(config)# errdisable recovery cause l2ptguard</pre>	(Optional) Configures the recovery mechanism from a Layer 2 maximum-rate error so that the interface is reenabled and can try again. Errdisable recovery is disabled by default; when enabled, the default time interval is 300 seconds.

	Command or Action	Purpose
Step 12	l2protocol-tunnel cos <i>value</i> Example: Device(config)# l2protocol-tunnel cos 2	(Optional) Configures the CoS value for all tunneled Layer 2 PDUs. The range is 0 to 7; the default is the default CoS value for the interface. If none is configured, the default is 5.
Step 13	end Example: Device(config)# end	Returns to privileged EXEC mode.
Step 14	show l2protocol Example: Device)# show l2protocol	Displays the Layer 2 tunnel ports on the device, including the protocols configured, the thresholds, and the counters.
Step 15	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Configuring the Customer Device

Before you begin

For EtherChannels, you need to configure both the SP edge device and the customer devices for Layer 2 protocol tunneling.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface *interface-id***
4. **switchport trunk encapsulation dot1q**
5. **switchport mode trunk**
6. **udld port**
7. **channel-group *channel-group-number* mode desirable**
8. **exit**
9. **interface port-channel *port-channel number***
10. **shutdown**
11. **no shutdown**
12. **end**
13. **show l2protocol**

14. copy running-config startup-config

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	interface <i>interface-id</i> Example: Device(config)# interface gigabitethernet1/0/1	Specifies the interface connected to the phone, and enters interface configuration mode.
Step 4	switchport trunk encapsulation dot1q Example: Device(config)# switchport trunk encapsulation dot1q	Sets the trunking encapsulation format to IEEE 802.1Q.
Step 5	switchport mode trunk Example: Device(config-if)# switchport mode trunk	Enables trunking on the interface.
Step 6	udld port Example: Device(config-if)# udld port	Enables UDLD in normal mode on the interface.
Step 7	channel-group <i>channel-group-number</i> mode desirable Example: Device(config-if)# channel-group 25 mode desirable	Assigns the interface to a channel group, and specifies desirable for the PAgP mode.
Step 8	exit	Returns to global configuration mode.

	Command or Action	Purpose
	<p>Example:</p> <pre>Device(config-if)# exit</pre>	
Step 9	<p>interface port-channel port-channel <i>number</i></p> <p>Example:</p> <pre>Device(config)# interface port-channel port-channel 25</pre>	Enters port-channel interface mode.
Step 10	<p>shutdown</p> <p>Example:</p> <pre>Device(config)# shutdown</pre>	Shuts down the interface.
Step 11	<p>no shutdown</p> <p>Example:</p> <pre>Device(config)# no shutdown</pre>	Enables the interface.
Step 12	<p>end</p> <p>Example:</p> <pre>Device(config)# end</pre>	Returns to privileged EXEC mode.
Step 13	<p>show l2protocol</p> <p>Example:</p> <pre>Device# show l2protocol</pre>	Displays the Layer 2 tunnel ports on the device, including the protocols configured, the thresholds, and the counters.
Step 14	<p>copy running-config startup-config</p> <p>Example:</p> <pre>Device# copy running-config startup-config</pre>	<p>(Optional) Saves your entries in the configuration file.</p> <p>Note Use the no switchport mode trunk, the no udd enable, and the no channel group channel-group-number mode desirable interface configuration commands to return the interface to the default settings.</p>

Configuration Examples for IEEE 802.1Q and Layer 2 Protocol Tunneling

Example: Configuring an IEEE 802.1Q Tunneling Port

The following example shows how to configure an interface as a tunnel port, enable tagging of native VLAN packets, and verify the configuration. In this configuration, the VLAN ID for the customer connected to Gigabit Ethernet interface 7 on stack member 1 is VLAN 22.

```
Switch(config)# interface gigabitethernet1/0/7
Switch(config-if)# switchport access vlan 22
% Access VLAN does not exist. Creating vlan 22
Switch(config-if)# switchport mode dot1q-tunnel
Switch(config-if)# exit
Switch(config)# vlan dot1q tag native
Switch(config)# end
Switch# show dot1q-tunnel interface gigabitethernet1/0/7
Port
-----
Gi1/0/1Port
-----
Switch# show vlan dot1q tag native
dot1q native vlan tagging is enabled
```

Example: Configuring Layer 2 Protocol Tunneling

The following example shows how to configure Layer 2 protocol tunneling for CDP, STP, and VTP and to verify the configuration.

```
Switch(config)# interface gigabitethernet1/0/11
Switch(config-if)# l2protocol-tunnel cdp
Switch(config-if)# l2protocol-tunnel stp
Switch(config-if)# l2protocol-tunnel vtp
Switch(config-if)# l2protocol-tunnel shutdown-threshold 1500
Switch(config-if)# l2protocol-tunnel drop-threshold 1000
Switch(config-if)# exit
Switch(config)# l2protocol-tunnel cos 7
Switch(config)# end
Switch# show l2protocol
```

```
COS for Encapsulated Packets: 7
Port Protocol Shutdown Drop Encapsulation Decapsulation Drop
Threshold Threshold Counter Counter Counter
-----
Gi0/11 cdp 1500 1000 2288 2282 0
stp 1500 1000 116 13 0
vtp 1500 1000 3 67 0
pagp ---- ---- 0 0 0
lacp ---- ---- 0 0 0
udld ---- ---- 0 0 0
```

Examples: Configuring the SP Edge and Customer Switches

This example shows how to configure the SP edge switch 1 and edge switch 2. VLANs 17, 18, 19, and 20 are the access VLANs, Fast Ethernet interfaces 1 and 2 are point-to-point tunnel ports with PAgP and UDLD enabled, the drop threshold is 1000, and Fast Ethernet interface 3 is a trunk port.

SP edge switch 1 configuration:

```
Switch(config)# interface gigabitethernet1/0/1
Switch(config-if)# switchport access vlan 17
Switch(config-if)# switchport mode dot1q-tunnel
Switch(config-if)# l2protocol-tunnel point-to-point pagp
Switch(config-if)# l2protocol-tunnel point-to-point udld
Switch(config-if)# l2protocol-tunnel drop-threshold point-to-point pagp 1000
Switch(config-if)# exit
Switch(config)# interface gigabitethernet1/0/2
Switch(config-if)# switchport access vlan 18
Switch(config-if)# switchport mode dot1q-tunnel
Switch(config-if)# l2protocol-tunnel point-to-point pagp
Switch(config-if)# l2protocol-tunnel point-to-point udld
Switch(config-if)# l2protocol-tunnel drop-threshold point-to-point pagp 1000
Switch(config-if)# exit
Switch(config)# interface gigabitethernet1/0/3
Switch(config-if)# switchport trunk encapsulation isl
Switch(config-if)# switchport mode trunk
```

SP edge switch 2 configuration:

```
Switch(config)# interface gigabitethernet1/0/1
Switch(config-if)# switchport access vlan 19
Switch(config-if)# switchport mode dot1q-tunnel
Switch(config-if)# l2protocol-tunnel point-to-point pagp
Switch(config-if)# l2protocol-tunnel point-to-point udld
Switch(config-if)# l2protocol-tunnel drop-threshold point-to-point pagp 1000
Switch(config-if)# exit
Switch(config)# interface gigabitethernet1/0/2
Switch(config-if)# switchport access vlan 20
Switch(config-if)# switchport mode dot1q-tunnel
Switch(config-if)# l2protocol-tunnel point-to-point pagp
Switch(config-if)# l2protocol-tunnel point-to-point udld
Switch(config-if)# l2protocol-tunnel drop-threshold point-to-point pagp 1000
Switch(config-if)# exit
Switch(config)# interface gigabitethernet1/0/3
Switch(config-if)# switchport trunk encapsulation isl
Switch(config-if)# switchport mode trunk
```

This example shows how to configure the customer switch at Site 1. Fast Ethernet interfaces 1, 2, 3, and 4 are set for IEEE 802.1Q trunking, UDLD is enabled, EtherChannel group 1 is enabled, and the port channel is shut down and then enabled to activate the EtherChannel configuration.

```
Switch(config)# interface gigabitethernet1/0/1
Switch(config-if)# switchport trunk encapsulation dot1q
Switch(config-if)# switchport mode trunk
Switch(config-if)# udld enable
Switch(config-if)# channel-group 1 mode desirable
Switch(config-if)# exit
Switch(config)# interface gigabitethernet1/0/2
```

```

Switch(config-if)# switchport trunk encapsulation dot1q
Switch(config-if)# switchport mode trunk
Switch(config-if)# uddld enable
Switch(config-if)# channel-group 1 mode desirable
Switch(config-if)# exit
Switch(config)# interface gigabitethernet1/0/3
Switch(config-if)# switchport trunk encapsulation dot1q
Switch(config-if)# switchport mode trunk
Switch(config-if)# uddld enable
Switch(config-if)# channel-group 1 mode desirable
Switch(config-if)# exit
Switch(config)# interface gigabitethernet1/0/4
Switch(config-if)# switchport trunk encapsulation dot1q
Switch(config-if)# switchport mode trunk
Switch(config-if)# uddld enable
Switch(config-if)# channel-group 1 mode desirable
Switch(config-if)# exit
Switch(config)# interface port-channel 1
Switch(config-if)# shutdown
Switch(config-if)# no shutdown
Switch(config-if)# exit

```

Monitoring Tunneling Status

The following table describes the commands used to monitor tunneling status.

Table 175: Commands for Monitoring Tunneling

Command	Purpose
clear l2protocol-tunnel counters	Clears the protocol counters on Layer 2 protocol tunneling ports.
show dot1q-tunnel	Displays IEEE 802.1Q tunnel ports on the device.
show dot1q-tunnel interface <i>interface-id</i>	Verifies if a specific interface is a tunnel port.
show l2protocol-tunnel	Displays information about Layer 2 protocol tunneling ports.
show errdisable recovery	Verifies if the recovery timer from a Layer 2 protocol-tunnel error disable state is enabled.
show l2protocol-tunnel interface <i>interface-id</i>	Displays information about a specific Layer 2 protocol tunneling port.
show l2protocol-tunnel summary	Displays only Layer 2 protocol summary information.
show vlan dot1q tag native	Displays the status of native VLAN tagging on the device.

Where to Go Next

You can configure the following:

- VTP
- VLANs
- VLAN Trunking
- Private VLANs
- VLAN Membership Policy Server (VMPS)
- Voice VLANs

Additional References

Related Documents

Related Topic	Document Title
For complete syntax and usage information for the commands used in this chapter.	<i>Catalyst 2960-XR Switch VLAN Management Command Reference</i>

Standards and RFCs

Standard/RFC	Title
—	—

MIBs

MIB	MIBs Link
All the supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
<p>The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies.</p> <p>To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds.</p> <p>Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.</p>	http://www.cisco.com/support

Feature History and Information for Tunneling

Release	Modification
Cisco IOS Release 15.0(2)EX1	This feature was introduced.



CHAPTER 90

Configuring Voice VLANs

- [Prerequisites for Voice VLANs, on page 1801](#)
- [Restrictions for Voice VLANs, on page 1801](#)
- [Information About Voice VLAN, on page 1802](#)
- [How to Configure Voice VLAN, on page 1804](#)
- [Monitoring Voice VLAN, on page 1808](#)
- [Configuration Examples for Voice VLANs, on page 1808](#)
- [Where to Go Next, on page 1809](#)
- [Additional References, on page 1809](#)
- [Feature History and Information for Voice VLAN, on page 1810](#)

Prerequisites for Voice VLANs

The following are the prerequisites for voice VLANs:

- Voice VLAN configuration is only supported on switch access ports; voice VLAN configuration is not supported on trunk ports.



Note Trunk ports can carry any number of voice VLANs, similar to regular VLANs. The configuration of voice VLANs is not supported on trunk ports.

- Before you enable voice VLAN, we recommend that you enable QoS on the switch by entering the **mls qos** global configuration command and configure the port trust state to trust by entering the **mls qos trust cos** interface configuration command. If you use the auto-QoS feature, these settings are automatically configured.
- You must enable CDP on the switch port connected to the Cisco IP Phone to send the configuration to the phone. (CDP is globally enabled by default on all switch interfaces.)

Restrictions for Voice VLANs

The following are the restrictions for voice VLANs:

- Do not configure voice VLAN on private VLAN ports.

- You cannot configure static secure MAC addresses in the voice VLAN.

Information About Voice VLAN

Voice VLANs

The voice VLAN feature enables access ports to carry IP voice traffic from an IP phone. When the device is connected to a Cisco 7960 IP Phone, the phone sends voice traffic with Layer 3 IP precedence and Layer 2 class of service (CoS) values, which are both set to 5 by default. Because the sound quality of an IP phone call can deteriorate if the data is unevenly sent, the device supports quality of service (QoS) based on IEEE 802.1p CoS. QoS uses classification and scheduling to send network traffic from the device in a predictable manner.

The Cisco 7960 IP Phone is a configurable device, and you can configure it to forward traffic with an IEEE 802.1p priority. You can configure the device to trust or override the traffic priority assigned by a Cisco IP Phone.

Cisco IP Phone Voice Traffic

You can configure an access port with an attached Cisco IP Phone to use one VLAN for voice traffic and another VLAN for data traffic from a device attached to the phone. You can configure access ports on the device to send Cisco Discovery Protocol (CDP) packets that instruct an attached phone to send voice traffic to the device in any of these ways:

- In the voice VLAN tagged with a Layer 2 CoS priority value
- In the access VLAN tagged with a Layer 2 CoS priority value
- In the access VLAN, untagged (no Layer 2 CoS priority value)



Note In all configurations, the voice traffic carries a Layer 3 IP precedence value (the default is 5 for voice traffic and 3 for voice control traffic).

Cisco IP Phone Data Traffic

The device can also process tagged data traffic (traffic in IEEE 802.1Q or IEEE 802.1p frame types) from the device attached to the access port on the Cisco IP Phone. You can configure Layer 2 access ports on the device to send CDP packets that instruct the attached phone to configure the phone access port in one of these modes:

- In trusted mode, all traffic received through the access port on the Cisco IP Phone passes through the phone unchanged.
- In untrusted mode, all traffic in IEEE 802.1Q or IEEE 802.1p frames received through the access port on the Cisco IP Phone receive a configured Layer 2 CoS value. The default Layer 2 CoS value is 0. Untrusted mode is the default.



Note Untagged traffic from the device attached to the Cisco IP Phone passes through the phone unchanged, regardless of the trust state of the access port on the phone.

Voice VLAN Configuration Guidelines

- Because a Cisco 7960 IP Phone also supports a connection to a PC or other device, a port connecting the device to a Cisco IP Phone can carry mixed traffic. You can configure a port to decide how the Cisco IP Phone carries voice traffic and data traffic.
- The voice VLAN should be present and active on the device for the IP phone to correctly communicate on the voice VLAN. Use the **show vlan** privileged EXEC command to see if the VLAN is present (listed in the display). If the VLAN is not listed, create the voice VLAN.
- The Power over Ethernet (PoE) devices are capable of automatically providing power to Cisco pre-standard and IEEE 802.3af-compliant powered devices if they are not being powered by an AC power source.
- The Port Fast feature is automatically enabled when voice VLAN is configured. When you disable voice VLAN, the Port Fast feature is not automatically disabled.
- If the Cisco IP Phone and a device attached to the phone are in the same VLAN, they must be in the same IP subnet. These conditions indicate that they are in the same VLAN:
 - They both use IEEE 802.1p or untagged frames.
 - The Cisco IP Phone uses IEEE 802.1p frames, and the device uses untagged frames.
 - The Cisco IP Phone uses untagged frames, and the device uses IEEE 802.1p frames.
 - The Cisco IP Phone uses IEEE 802.1Q frames, and the voice VLAN is the same as the access VLAN.
- The Cisco IP Phone and a device attached to the phone cannot communicate if they are in the same VLAN and subnet but use different frame types because traffic in the same subnet is not routed (routing would eliminate the frame type difference).
- Voice VLAN ports can also be these port types:
 - Dynamic access port.
 - IEEE 802.1x authenticated port.



Note If you enable IEEE 802.1x on an access port on which a voice VLAN is configured and to which a Cisco IP Phone is connected, the phone loses connectivity to the device for up to 30 seconds.

- Protected port.
- A source or destination port for a SPAN or RSPAN session.
- Secure port.



Note When you enable port security on an interface that is also configured with a voice VLAN, you must set the maximum allowed secure addresses on the port to two plus the maximum number of secure addresses allowed on the access VLAN. When the port is connected to a Cisco IP Phone, the phone requires up to two MAC addresses. The phone address is learned on the voice VLAN and might also be learned on the access VLAN. Connecting a PC to the phone requires additional MAC addresses.

Default Voice VLAN Configuration

The voice VLAN feature is disabled by default.

When the voice VLAN feature is enabled, all untagged traffic is sent according to the default CoS priority of the port.

The CoS value is not trusted for IEEE 802.1p or IEEE 802.1Q tagged traffic.

How to Configure Voice VLAN

Configuring Cisco IP Phone Voice Traffic

You can configure a port connected to the Cisco IP Phone to send CDP packets to the phone to configure the way in which the phone sends voice traffic. The phone can carry voice traffic in IEEE 802.1Q frames for a specified voice VLAN with a Layer 2 CoS value. It can use IEEE 802.1p priority tagging to give voice traffic a higher priority and forward all voice traffic through the native (access) VLAN. The Cisco IP Phone can also send untagged voice traffic or use its own configuration to send voice traffic in the access VLAN. In all configurations, the voice traffic carries a Layer 3 IP precedence value (the default is 5).

SUMMARY STEPS

1. **configure terminal**
2. **interface** *interface-id*
3. **mls qos trust cos**
4. **switchport voice vlan** {*vlan-id* | **dot1p** | **none** | **untagged** }
5. **end**
6. Use one of the following:
 - **show interfaces** *interface-id* **switchport**
 - **show running-config interface** *interface-id*
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 2	interface interface-id Example: Device(config)# interface gigabitethernet1/0/1	Specifies the interface connected to the phone, and enters interface configuration mode.
Step 3	mls qos trust cos Example: Device(config-if)# mls qos trust cos	Configures the interface to classify incoming traffic packets by using the packet CoS value. For untagged packets, the port default CoS value is used. Note Before configuring the port trust state, you must first globally enable QoS by using the mls qos global configuration command.
Step 4	switchport voice vlan {vlan-id dot1p none untagged } Example: Device(config-if)# switchport voice vlan 125	Configures how the Cisco IP Phone carries voice traffic: <ul style="list-style-type: none"> • vlan-id—Configures the phone to forward all voice traffic through the specified VLAN. By default, the Cisco IP Phone forwards the voice traffic with an IEEE 802.1Q priority of 5. Valid VLAN IDs are 1 to 4094. • dot1p—Configures the switch to accept voice and data IEEE 802.1p priority frames tagged with VLAN ID 0 (the native VLAN). By default, the switch drops all voice and data traffic tagged with VLAN 0. If configured for 802.1p the Cisco IP Phone forwards the traffic with an IEEE 802.1p priority of 5. • none—Allows the phone to use its own configuration to send untagged voice traffic. • untagged—Configures the phone to send untagged voice traffic. Note Before configuring the switch port to detect and recognize a Cisco IP phone, confirm that the phone is powered by PoE. The configuration fails when power is provided by an AC source.
Step 5	end Example:	Returns to privileged EXEC mode.

	Command or Action	Purpose
	Device(config-if)# end	
Step 6	Use one of the following: <ul style="list-style-type: none"> • show interfaces <i>interface-id</i> switchport • show running-config interface <i>interface-id</i> Example: Device# show interfaces gigabitethernet1/0/1 switchport or Device# show running-config interface gigabitethernet1/0/1	Verifies your voice VLAN entries or your QoS and voice VLAN entries.
Step 7	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Configuring the Priority of Incoming Data Frames

You can connect a PC or other data device to a Cisco IP Phone port. To process tagged data traffic (in IEEE 802.1Q or IEEE 802.1p frames), you can configure the device to send CDP packets to instruct the phone how to send data packets from the device attached to the access port on the Cisco IP Phone. The PC can generate packets with an assigned CoS value. You can configure the phone to not change (trust) or to override (not trust) the priority of frames arriving on the phone port from connected devices.

Follow these steps to set the priority of data traffic received from the non-voice port on the Cisco IP Phone:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **switchport priority extend** {*cos value* | **trust**}
5. **end**
6. **show interfaces** *interface-id* **switchport**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable	Enables privileged EXEC mode.

	Command or Action	Purpose
	Example: Device> enable	<ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	interface interface-id Example: Device(config)# interface gigabitethernet1/0/1	Specifies the interface connected to the Cisco IP Phone, and enters interface configuration mode.
Step 4	switchport priority extend {cos value trust} Example: Device(config-if)# switchport priority extend trust	Sets the priority of data traffic received from the Cisco IP Phone access port: <ul style="list-style-type: none"> • cos value—Configures the phone to override the priority received from the PC or the attached device with the specified CoS value. The value is a number from 0 to 7, with 7 as the highest priority. The default priority is cos 0. • trust—Configures the phone access port to trust the priority received from the PC or the attached device. <p>Note To return the port to its default setting, use the no switchport priority extend interface configuration command.</p>
Step 5	end Example: Device(config-if)# end	Returns to privileged EXEC mode.
Step 6	show interfaces interface-id switchport Example: Device# show interfaces gigabitethernet1/0/1 switchport	Verifies your entries.
Step 7	copy running-config startup-config Example:	(Optional) Saves your entries in the configuration file.

	Command or Action	Purpose
	Device# <code>copy running-config startup-config</code>	

Monitoring Voice VLAN

To display voice VLAN configuration for an interface, use the **show interfaces *interface-id* switchport** privileged EXEC command.

Configuration Examples for Voice VLANs

Example: Configuring Cisco IP Phone Voice Traffic

This example shows how to configure a port connected to a Cisco IP Phone to use the CoS value to classify incoming traffic and to accept voice and data priority traffic tagged with VLAN ID 0:

```
Switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Switch(config)# interface gigabitethernet1/0/1
Switch(config-if)# mls qos trust cos
Switch(config-if)# switchport voice vlan dot1p
Switch(config-if)# end
```

To return the port to its default setting, use the **no switchport voice vlan** interface configuration command.

This example shows how to enable switch port voice detect on a Cisco IP Phone:

```
Switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Switch(config)# interface gigabitethernet 1/0/1
Switch(config-if)# switchport voice?
detect detection enhancement keyword
vlan VLAN for voice traffic

Switch(config-if)# switchport voice detect?
cisco-phone Cisco IP Phone

Switch(config-if)# switchport voice detect cisco-phone?
full-duplex Cisco IP Phone

Switch(config-if)# switchport voice detect cisco-phone full-duplex
full-duplex full duplex keyword

Switch(config-if)# end
```

This example shows how to disable switchport voice detect on a Cisco IP Phone:

```
Switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Switch(config)# interface gigabitethernet 1/0/1
```

```
Switch(config-if)# no switchport voice detect cisco-phone
Switch(config-if)# no switchport voice detect cisco-phone full-duplex
```

Example: Configuring a Port Connected to an IP Phone Not to Change Frame Priority

This example shows how to configure a port connected to a Cisco IP Phone to not change the priority of frames received from the PC or the attached device:

```
Switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Switch(config)# interface gigabitethernet1/0/1
Switch(config-if)# switchport priority extend trust
Switch(config-if)# end
```

Where to Go Next

After configuring voice VLANs, you can configure the following:

- VTP
- VLANs
- VLAN trunking
- Private VLANs
- VLAN Membership Policy Server (VMPS)
- Tunneling

Additional References

Related Documents

Related Topic	Document Title
For complete syntax and usage information for the commands used in this chapter.	<i>Catalyst 2960-XR Switch VLAN Management Command Reference</i>

Standards and RFCs

Standard/RFC	Title
—	—

MIBs

MIB	MIBs Link
All the supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
<p>The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies.</p> <p>To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds.</p> <p>Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.</p>	http://www.cisco.com/support

Feature History and Information for Voice VLAN

Release	Modification
Cisco IOS Release 15.0(2)EX1	This feature was introduced.



PART **XV**

System Management

- [Administering the System, on page 1813](#)
- [Performing Device Setup Configuration, on page 1847](#)
- [Configuring AVC with DNS-AS, on page 1875](#)
- [Configuring SDM Templates, on page 1899](#)
- [Configuring System Message Logging and Smart Logging, on page 1907](#)
- [Configuring Online Diagnostics, on page 1927](#)
- [Troubleshooting the Software Configuration, on page 1939](#)
- [Information About Licensing, on page 1969](#)



CHAPTER 91

Administering the System

- [Information About Administering the Device](#), on page 1813
- [How to Administer the Device](#), on page 1820
- [Monitoring and Maintaining Administration of the Device](#), on page 1841
- [Configuration Examples for Device Administration](#), on page 1842
- [Additional References for Switch Administration](#), on page 1844
- [Feature History and Information for Device Administration](#), on page 1845

Information About Administering the Device

System Time and Date Management

You can manage the system time and date on your device using automatic configuration methods (RTC and NTP), or manual configuration methods.



Note For complete syntax and usage information for the commands used in this section, see the *Cisco IOS Configuration Fundamentals Command Reference* on Cisco.com.

System Clock

The basis of the time service is the system clock. This clock runs from the moment the system starts up and keeps track of the date and time.

The system clock can then be set from these sources:

- RTC
- NTP
- Manual configuration

The system clock can provide time to these services:

- User **show** commands

- Logging and debugging messages

The system clock keeps track of time internally based on Coordinated Universal Time (UTC), also known as Greenwich Mean Time (GMT). You can configure information about the local time zone and summer time (daylight saving time) so that the time appears correctly for the local time zone.

The system clock keeps track of whether the time is *authoritative* or not (that is, whether it has been set by a time source considered to be authoritative). If it is not authoritative, the time is available only for display purposes and is not redistributed.

Real Time Clock

A real-time clock (RTC) keeps track of the current time on the switch. The switch is shipped to you with RTC set to GMT time until you reconfigure clocking parameters.

The benefits of an RTC are:

- RTC is battery-powered.
- System time is retained during power outage and at system reboot.

The RTC and NTP clocks are integrated on the switch. When NTP is enabled, the RTC time is periodically synchronized to the NTP clock to maintain accuracy.

Network Time Protocol

The NTP is designed to time-synchronize a network of devices. NTP runs over User Datagram Protocol (UDP), which runs over IP. NTP is documented in RFC 1305.

An NTP network usually gets its time from an authoritative time source, such as a radio clock or an atomic clock attached to a time server. NTP then distributes this time across the network. NTP is extremely efficient; no more than one packet per minute is necessary to synchronize two devices to within a millisecond of one another.

NTP uses the concept of a *stratum* to describe how many NTP hops away a device is from an authoritative time source. A stratum 1 time server has a radio or atomic clock directly attached, a stratum 2 time server receives its time through NTP from a stratum 1 time server, and so on. A device running NTP automatically chooses as its time source the device with the lowest stratum number with which it communicates through NTP. This strategy effectively builds a self-organizing tree of NTP speakers.

NTP avoids synchronizing to a device whose time might not be accurate by never synchronizing to a device that is not synchronized. NTP also compares the time reported by several devices and does not synchronize to a device whose time is significantly different than the others, even if its stratum is lower.

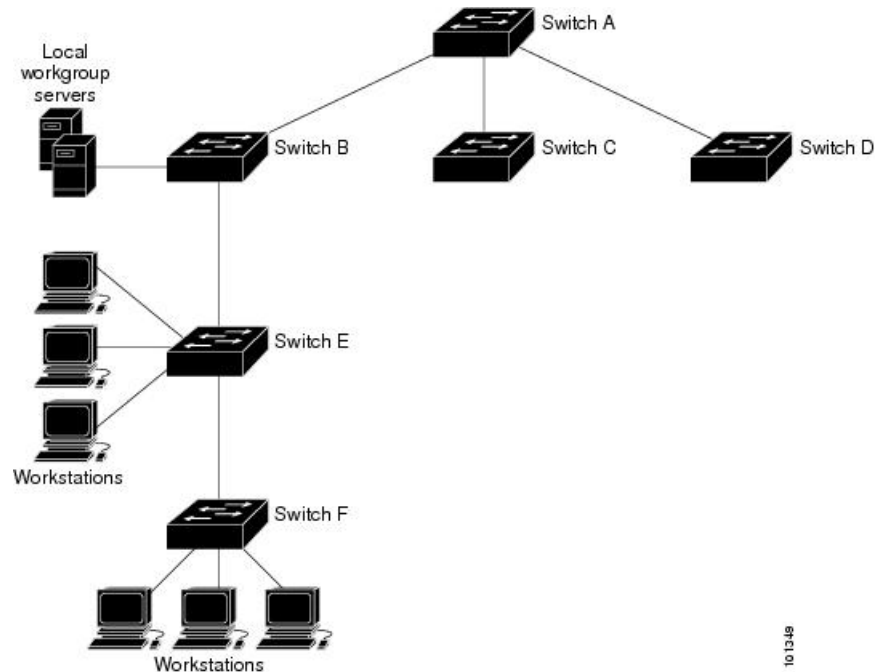
The communications between devices running NTP (known as associations) are usually statically configured; each device is given the IP address of all devices with which it should form associations. Accurate timekeeping is possible by exchanging NTP messages between each pair of devices with an association. However, in a LAN environment, NTP can be configured to use IP broadcast messages instead. This alternative reduces configuration complexity because each device can simply be configured to send or receive broadcast messages. However, in that case, information flow is one-way only.

The time kept on a device is a critical resource; you should use the security features of NTP to avoid the accidental or malicious setting of an incorrect time. Two mechanisms are available: an access list-based restriction scheme and an encrypted authentication mechanism.

Cisco's implementation of NTP does not support stratum 1 service; it is not possible to connect to a radio or atomic clock. We recommend that the time service for your network be derived from the public NTP servers available on the IP Internet.

The figure below shows a typical network example using NTP. Device A is the NTP primary (formerly known as NTP primary), with the **Device B, C, and D** configured in NTP server mode, in server association with Device A. Device E is configured as an NTP peer to the upstream and downstream Device, Device B and Device F, respectively.

Figure 137: Typical NTP Network Configuration



If the network is isolated from the Internet, Cisco's implementation of NTP allows a device to act as if it is synchronized through NTP, when in fact it has learned the time by using other means. Other devices then synchronize to that device through NTP.

When multiple sources of time are available, NTP is always considered to be more authoritative. NTP time overrides the time set by any other method.

Several manufacturers include NTP software for their host systems, and a publicly available version for systems running UNIX and its various derivatives is also available. This software allows host systems to be time-synchronized as well.

NTP Stratum

NTP uses the concept of a *stratum* to describe how many NTP hops away a device is from an authoritative time source. A stratum 1 time server has a radio or atomic clock directly attached, a stratum 2 time server receives its time through NTP from a stratum 1 time server, and so on. A device running NTP automatically chooses as its time source the device with the lowest stratum number with which it communicates through NTP. This strategy effectively builds a self-organizing tree of NTP speakers.

NTP avoids synchronizing to a device whose time might not be accurate by never synchronizing to a device that is not synchronized. NTP also compares the time reported by several devices and does not synchronize to a device whose time is significantly different than the others, even if its stratum is lower.

NTP Associations

The communications between devices running NTP (known as *associations*) are usually statically configured; each device is given the IP address of all devices with which it should form associations. Accurate timekeeping is possible by exchanging NTP messages between each pair of devices with an association. However, in a LAN environment, NTP can be configured to use IP broadcast messages instead. This alternative reduces configuration complexity because each device can simply be configured to send or receive broadcast messages. However, in that case, information flow is one-way only.

NTP Security

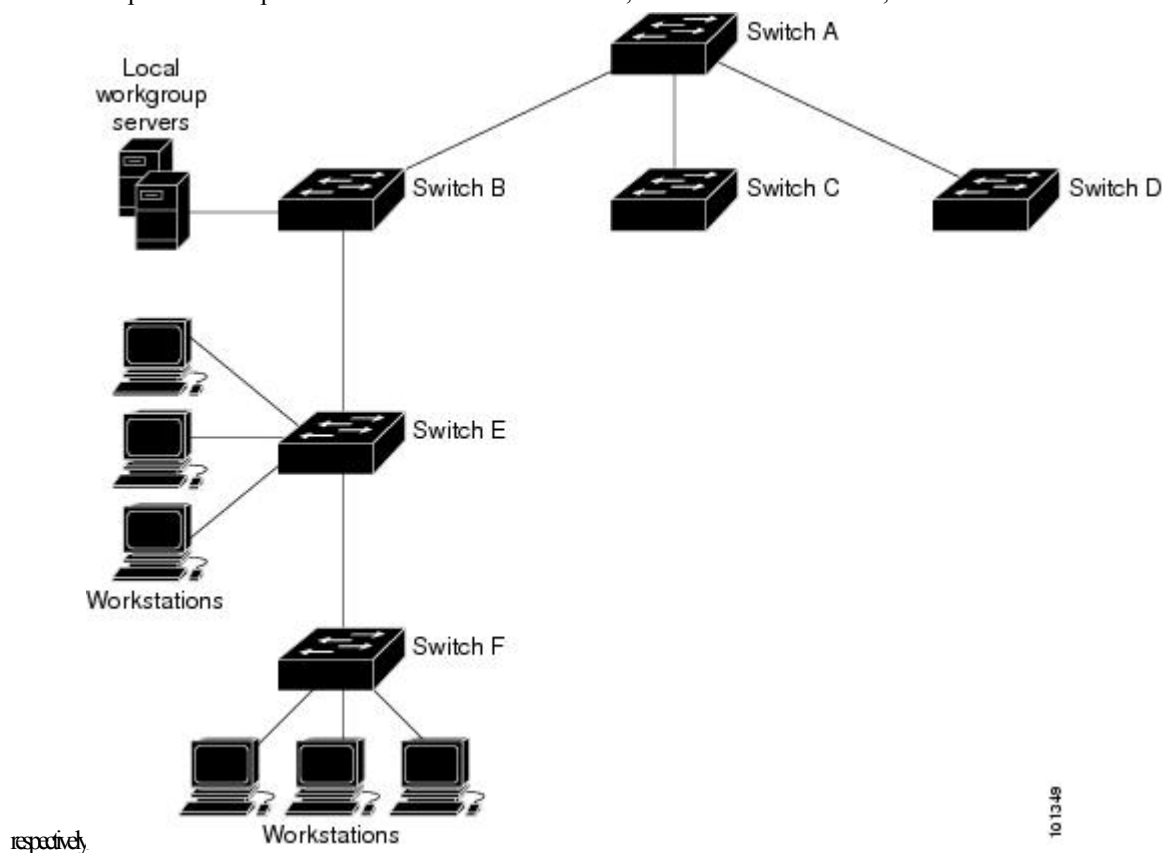
The time kept on a device is a critical resource; you should use the security features of NTP to avoid the accidental or malicious setting of an incorrect time. Two mechanisms are available: an access list-based restriction scheme and an encrypted authentication mechanism.

NTP Implementation

Implementation of NTP does not support stratum 1 service; it is not possible to connect to a radio or atomic clock. We recommend that the time service for your network be derived from the public NTP servers available on the IP Internet.

Figure 138: Typical NTP Network Configuration

The following figure shows a typical network example using NTP. Switch A is the NTP primary, with the Switch B, C, and D configured in NTP server mode, in server association with Switch A. Switch E is configured as an NTP peer to the upstream and downstream switches, Switch B and Switch F,



If the network is isolated from the Internet, NTP allows a device to act as if it is synchronized through NTP, when in fact it has learned the time by using other means. Other devices then synchronize to that device through NTP.

When multiple sources of time are available, NTP is always considered to be more authoritative. NTP time overrides the time set by any other method.

Several manufacturers include NTP software for their host systems, and a publicly available version for systems running UNIX and its various derivatives is also available. This software allows host systems to be time-synchronized as well.

NTP Version 4

NTP version 4 is implemented on the device. NTPv4 is an extension of NTP version 3. NTPv4 supports both IPv4 and IPv6 and is backward-compatible with NTPv3.

NTPv4 provides these capabilities:

- Support for IPv6.
- Improved security compared to NTPv3. The NTPv4 protocol provides a security framework based on public key cryptography and standard X509 certificates.
- Automatic calculation of the time-distribution hierarchy for a network. Using specific multicast groups, NTPv4 automatically configures the hierarchy of the servers to achieve the best time accuracy for the lowest bandwidth cost. This feature leverages site-local IPv6 multicast addresses.

For details about configuring NTPv4, see the *Implementing NTPv4 in IPv6* chapter of the *Cisco IOS IPv6 Configuration Guide, Release 12.4T*.

System Name and Prompt

You configure the system name on the Device to identify it. By default, the system name and prompt are *Switch*.

If you have not configured a system prompt, the first 20 characters of the system name are used as the system prompt. A greater-than symbol [`>`] is appended. The prompt is updated whenever the system name changes.

For complete syntax and usage information for the commands used in this section, see the *Cisco IOS Configuration Fundamentals Command Reference, Release 12.4* and the *Cisco IOS IP Command Reference, Volume 2 of 3: Routing Protocols, Release 12.4*.

Stack System Name and Prompt

If you are accessing a stack member through the active stack, you must use the **session** *stack-member-number* privileged EXEC command. The stack member number range is from 1 through 8. When you use this command, the stack member number is appended to the system prompt. For example, *Switch-2#* is the prompt in privileged EXEC mode for stack member 2, and the system prompt for the switch stack is *Switch*.

Default System Name and Prompt Configuration

The default switch system name and prompt is *Switch*.

DNS

The DNS protocol controls the Domain Name System (DNS), a distributed database with which you can map hostnames to IP addresses. When you configure DNS on your device, you can substitute the hostname for the IP address with all IP commands, such as **ping**, **telnet**, **connect**, and related Telnet support operations.

IP defines a hierarchical naming scheme that allows a device to be identified by its location or domain. Domain names are pieced together with periods (.) as the delimiting characters. For example, Cisco Systems is a commercial organization that IP identifies by a *com* domain name, so its domain name is *cisco.com*. A specific device in this domain, for example, the File Transfer Protocol (FTP) system is identified as *ftp.cisco.com*.

To keep track of domain names, IP has defined the concept of a domain name server, which holds a cache (or database) of names mapped to IP addresses. To map domain names to IP addresses, you must first identify the hostnames, specify the name server that is present on your network, and enable the DNS.

Default DNS Settings

Table 176: Default DNS Settings

Feature	Default Setting
DNS enable state	Enabled.
DNS default domain name	None configured.
DNS servers	No name server addresses are configured.

Login Banners

You can configure a message-of-the-day (MOTD) and a login banner. The MOTD banner is displayed on all connected terminals at login and is useful for sending messages that affect all network users (such as impending system shutdowns).

The login banner is also displayed on all connected terminals. It appears after the MOTD banner and before the login prompts.



Note For complete syntax and usage information for the commands used in this section, see the *Cisco IOS Configuration Fundamentals Command Reference, Release 12.4*.

Default Banner Configuration

The MOTD and login banners are not configured.

MAC Address Table

The MAC address table contains address information that the device uses to forward traffic between ports. All MAC addresses in the address table are associated with one or more ports. The address table includes these types of addresses:

- Dynamic address—A source MAC address that the device learns and then ages when it is not in use.

- Static address—A manually entered unicast address that does not age and that is not lost when the device resets.

The address table lists the destination MAC address, the associated VLAN ID, and port number associated with the address and the type (static or dynamic).



Note For complete syntax and usage information for the commands used in this section, see the command reference for this release.

MAC Address Table Creation

With multiple MAC addresses supported on all ports, you can connect any port on the device to other network devices. The device provides dynamic addressing by learning the source address of packets it receives on each port and adding the address and its associated port number to the address table. As devices are added or removed from the network, the device updates the address table, adding new dynamic addresses and aging out those that are not in use.

The aging interval is globally configured. However, the device maintains an address table for each VLAN, and STP can accelerate the aging interval on a per-VLAN basis.

The device sends packets between any combination of ports, based on the destination address of the received packet. Using the MAC address table, the device forwards the packet only to the port associated with the destination address. If the destination address is on the port that sent the packet, the packet is filtered and not forwarded. The device always uses the store-and-forward method: complete packets are stored and checked for errors before transmission.

MAC Addresses and VLANs

All addresses are associated with a VLAN. An address can exist in more than one VLAN and have different destinations in each. Unicast addresses, for example, could be forwarded to port 1 in VLAN 1 and ports 9, 10, and 1 in VLAN 5.

Each VLAN maintains its own logical address table. A known address in one VLAN is unknown in another until it is learned or statically associated with a port in the other VLAN.

When private VLANs are configured, address learning depends on the type of MAC address:

- Dynamic MAC addresses learned in one VLAN of a private VLAN are replicated in the associated VLANs. For example, a MAC address learned in a private-VLAN secondary VLAN is replicated in the primary VLAN.
- Static MAC addresses configured in a primary or secondary VLAN are not replicated in the associated VLANs. When you configure a static MAC address in a private VLAN primary or secondary VLAN, you should also configure the same static MAC address in all associated VLANs.

MAC Addresses and Device Stacks

The MAC address tables on all stack members are synchronized. At any given time, each stack member has the same copy of the address tables for each VLAN. When an address ages out, the address is removed from the address tables on all stack members. When a Device joins a switch stack, that Device receives the addresses for each VLAN learned on the other stack members. When a stack member leaves the switch stack, the remaining stack members age out or remove all addresses learned by the former stack member.

Default MAC Address Table Settings

The following table shows the default settings for the MAC address table.

Table 177: Default Settings for the MAC Address

Feature	Default Setting
Aging time	300 seconds
Dynamic addresses	Automatically learned
Static addresses	None configured

ARP Table Management

To communicate with a device (over Ethernet, for example), the software first must learn the 48-bit MAC address or the local data link address of that device. The process of learning the local data link address from an IP address is called *address resolution*.

The Address Resolution Protocol (ARP) associates a host IP address with the corresponding media or MAC addresses and the VLAN ID. Using an IP address, ARP finds the associated MAC address. When a MAC address is found, the IP-MAC address association is stored in an ARP cache for rapid retrieval. Then the IP datagram is encapsulated in a link-layer frame and sent over the network. Encapsulation of IP datagrams and ARP requests and replies on IEEE 802 networks other than Ethernet is specified by the Subnetwork Access Protocol (SNAP). By default, standard Ethernet-style ARP encapsulation (represented by the **arpa** keyword) is enabled on the IP interface.

ARP entries added manually to the table do not age and must be manually removed.

For CLI procedures, see the Cisco IOS Release 12.4 documentation on *Cisco.com*.

How to Administer the Device

Configuring the Time and Date Manually

System time remains accurate through restarts and reboot, however, you can manually configure the time and date after the system is restarted.

We recommend that you use manual configuration only when necessary. If you have an outside source to which the device can synchronize, you do not need to manually set the system clock.

Setting the System Clock

If you have an outside source on the network that provides time services, such as an NTP server, you do not need to manually set the system clock.

Follow these steps to set the system clock:

SUMMARY STEPS

1. **enable**
2. Use one of the following:
 - **clock set** *hh:mm:ss day month year*
 - **clock set** *hh:mm:ss month day year*

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	Use one of the following: <ul style="list-style-type: none"> • clock set <i>hh:mm:ss day month year</i> • clock set <i>hh:mm:ss month day year</i> Example: Device# clock set 13:32:00 23 March 2013	Manually set the system clock using one of these formats: <ul style="list-style-type: none"> • <i>hh:mm:ss</i>—Specifies the time in hours (24-hour format), minutes, and seconds. The time specified is relative to the configured time zone. • <i>day</i>—Specifies the day by date in the month. • <i>month</i>—Specifies the month by name. • <i>year</i>—Specifies the year (no abbreviation).

Configuring the Time Zone

Follow these steps to manually configure the time zone:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **clock timezone** *zone hours-offset [minutes-offset]*
4. **end**
5. **show running-config**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.

	Command or Action	Purpose
Step 2	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode.
Step 3	clock timezone zone hours-offset [minutes-offset] Example: Device(config)# <code>clock timezone AST -3 30</code>	Sets the time zone. Internal time is kept in Coordinated Universal Time (UTC), so this command is used only for display purposes and when the time is manually set. <ul style="list-style-type: none"> • <i>zone</i>—Enters the name of the time zone to be displayed when standard time is in effect. The default is UTC. • <i>hours-offset</i>—Enters the hours offset from UTC. • (Optional) <i>minutes-offset</i>—Enters the minutes offset from UTC. This available where the local time zone is a percentage of an hour different from UTC.
Step 4	end Example: Device(config)# <code>end</code>	Returns to privileged EXEC mode.
Step 5	show running-config Example: Device# <code>show running-config</code>	Verifies your entries.
Step 6	copy running-config startup-config Example: Device# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Configuring Summer Time (Daylight Saving Time)

To configure summer time (daylight saving time) in areas where it starts and ends on a particular day of the week each year, perform this task:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **clock summer-time zone date date month year hh:mm date month year hh:mm [offset]**
4. **clock summer-time zone recurring [week day month hh:mm week day month hh:mm [offset]]**

5. `end`
6. `show running-config`
7. `copy running-config startup-config`

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p>enable</p> <p>Example:</p> <pre>Device> enable</pre>	<p>Enables privileged EXEC mode.</p> <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	<p>configure terminal</p> <p>Example:</p> <pre>Device# configure terminal</pre>	<p>Enters global configuration mode.</p>
Step 3	<p>clock summer-time <i>zone</i> date <i>date month year hh:mm</i> <i>date month year hh:mm [offset]</i></p> <p>Example:</p> <pre>Device(config)# clock summer-time PDT date 10 March 2013 2:00 3 November 2013 2:00</pre>	<p>Configures summer time to start and end on specified days every year.</p>
Step 4	<p>clock summer-time <i>zone</i> recurring [<i>week day month</i> <i>hh:mm week day month hh:mm [offset]</i>]</p> <p>Example:</p> <pre>Device(config)# clock summer-time PDT recurring 10 March 2013 2:00 3 November 2013 2:00</pre>	<p>Configures summer time to start and end on the specified days every year. All times are relative to the local time zone. The start time is relative to standard time.</p> <p>The end time is relative to summer time. Summer time is disabled by default. If you specify clock summer-time <i>zone</i> recurring without parameters, the summer time rules default to the United States rules.</p> <p>If the starting month is after the ending month, the system assumes that you are in the southern hemisphere.</p> <ul style="list-style-type: none"> • <i>zone</i>—Specifies the name of the time zone (for example, PDT) to be displayed when summer time is in effect. • (Optional) <i>week</i>— Specifies the week of the month (1 to 4, first, or last). • (Optional) <i>day</i>—Specifies the day of the week (Sunday, Monday...). • (Optional) <i>month</i>—Specifies the month (January, February...).

	Command or Action	Purpose
		<ul style="list-style-type: none"> • (Optional) <i>hh:mm</i>—Specifies the time (24-hour format) in hours and minutes. • (Optional) <i>offset</i>—Specifies the number of minutes to add during summer time. The default is 60.
Step 5	end Example: <pre>Device(config)# end</pre>	Returns to privileged EXEC mode.
Step 6	show running-config Example: <pre>Device# show running-config</pre>	Verifies your entries.
Step 7	copy running-config startup-config Example: <pre>Device# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Follow these steps if summer time in your area does not follow a recurring pattern (configure the exact date and time of the next summer time events):

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **clock summer-time zone date** [*month date year hh:mm month date year hh:mm [offset]*] **or** **clock summer-time zone date** [*date month year hh:mm date month year hh:mm [offset]*]
4. **end**
5. **show running-config**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Device> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.

	Command or Action	Purpose
Step 2	<p>configure terminal</p> <p>Example:</p> <pre>Device# configure terminal</pre>	Enters global configuration mode.
Step 3	<p>clock summer-time zone date [<i>month date year hh:mm month date year hh:mm [offset]</i>] or clock summer-time zone date [<i>date month year hh:mm date month year hh:mm [offset]</i>]</p>	<p>Configures summer time to start on the first date and end on the second date.</p> <p>Summer time is disabled by default.</p> <ul style="list-style-type: none"> • For <i>zone</i>, specify the name of the time zone (for example, PDT) to be displayed when summer time is in effect. • (Optional) For <i>week</i>, specify the week of the month (1 to 5 or last). • (Optional) For <i>day</i>, specify the day of the week (Sunday, Monday...). • (Optional) For <i>month</i>, specify the month (January, February...). • (Optional) For <i>hh:mm</i>, specify the time (24-hour format) in hours and minutes. • (Optional) For <i>offset</i>, specify the number of minutes to add during summer time. The default is 60.
Step 4	<p>end</p> <p>Example:</p> <pre>Device(config)# end</pre>	Returns to privileged EXEC mode.
Step 5	<p>show running-config</p> <p>Example:</p> <pre>Device# show running-config</pre>	Verifies your entries.
Step 6	<p>copy running-config startup-config</p> <p>Example:</p> <pre>Device# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Configuring a System Name

Follow these steps to manually configure a system name:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **hostname *name***
4. **end**
5. **show running-config**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	hostname <i>name</i> Example: Device(config)# hostname remote-users	Configures a system name. When you set the system name, it is also used as the system prompt. The default setting is Switch. The name must follow the rules for ARPANET hostnames. They must start with a letter, end with a letter or digit, and have as interior characters only letters, digits, and hyphens. Names can be up to 63 characters.
Step 4	end Example: remote-users(config)# end remote-users#	Returns to privileged EXEC mode.
Step 5	show running-config Example: Device# show running-config	Verifies your entries.

	Command or Action	Purpose
Step 6	copy running-config startup-config Example: Device# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Setting Up DNS

If you use the device IP address as its hostname, the IP address is used and no DNS query occurs. If you configure a hostname that contains no periods (.), a period followed by the default domain name is appended to the hostname before the DNS query is made to map the name to an IP address. The default domain name is the value set by the **ip domain-name** global configuration command. If there is a period (.) in the hostname, the Cisco IOS software looks up the IP address without appending any default domain name to the hostname.

Follow these steps to set up your switch to use the DNS:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip domain-name** *name*
4. **ip name-server** *server-address1* [*server-address2* ... *server-address6*]
5. **ip domain-lookup** [*nsap* | *source-interface interface*]
6. **end**
7. **show running-config**
8. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> <code>enable</code>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode.
Step 3	ip domain-name <i>name</i> Example: Device(config)# <code>ip domain-name Cisco.com</code>	Defines a default domain name that the software uses to complete unqualified hostnames (names without a dotted-decimal domain name).

	Command or Action	Purpose
		<p>Do not include the initial period that separates an unqualified name from the domain name.</p> <p>At boot time, no domain name is configured; however, if the device configuration comes from a BOOTP or Dynamic Host Configuration Protocol (DHCP) server, then the default domain name might be set by the BOOTP or DHCP server (if the servers were configured with this information).</p>
Step 4	<p>ip name-server <i>server-address1</i> [<i>server-address2</i> ... <i>server-address6</i>]</p> <p>Example:</p> <pre>Device(config)# ip name-server 192.168.1.100 192.168.1.200 192.168.1.300</pre>	<p>Specifies the address of one or more name servers to use for name and address resolution.</p> <p>You can specify up to six name servers. Separate each server address with a space. The first server specified is the primary server. The device sends DNS queries to the primary server first. If that query fails, the backup servers are queried.</p>
Step 5	<p>ip domain-lookup [<i>nsap</i> <i>source-interface interface</i>]</p> <p>Example:</p> <pre>Device(config)# ip domain-lookup</pre>	<p>(Optional) Enables DNS-based hostname-to-address translation on your device. This feature is enabled by default.</p> <p>If your network devices require connectivity with devices in networks for which you do not control name assignment, you can dynamically assign device names that uniquely identify your devices by using the global Internet naming scheme (DNS).</p>
Step 6	<p>end</p> <p>Example:</p> <pre>Device(config)# end</pre>	<p>Returns to privileged EXEC mode.</p>
Step 7	<p>show running-config</p> <p>Example:</p> <pre>Device# show running-config</pre>	<p>Verifies your entries.</p>
Step 8	<p>copy running-config startup-config</p> <p>Example:</p> <pre>Device# copy running-config startup-config</pre>	<p>(Optional) Saves your entries in the configuration file.</p>

Configuring a Message-of-the-Day Login Banner

You can create a single or multiline message banner that appears on the screen when someone logs in to the device

Follow these steps to configure a MOTD login banner:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **banner motd *c message c***
4. **end**
5. **show running-config**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	banner motd <i>c message c</i> Example: Device(config)# banner motd # This is a secure site. Only authorized users are allowed. For access, contact technical support. #	Specifies the message of the day. <i>c</i> —Enters the delimiting character of your choice, for example, a pound sign (#), and press the Return key. The delimiting character signifies the beginning and end of the banner text. Characters after the ending delimiter are discarded. <i>message</i> —Enters a banner message up to 255 characters. You cannot use the delimiting character in the message.
Step 4	end Example: Device(config)# end	Returns to privileged EXEC mode.
Step 5	show running-config Example: Device# show running-config	Verifies your entries.

	Command or Action	Purpose
Step 6	copy running-config startup-config Example: <pre>Device# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Configuring a Login Banner

You can configure a login banner to be displayed on all connected terminals. This banner appears after the MOTD banner and before the login prompt.

Follow these steps to configure a login banner:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **banner login *c message c***
4. **end**
5. **show running-config**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Device> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: <pre>Device# configure terminal</pre>	Enters global configuration mode.
Step 3	banner login <i>c message c</i> Example: <pre>Device(config)# banner login \$ Access for authorized users only. Please enter your username and password. \$</pre>	Specifies the login message. <p><i>c</i>— Enters the delimiting character of your choice, for example, a pound sign (#), and press the Return key. The delimiting character signifies the beginning and end of the banner text. Characters after the ending delimiter are discarded.</p> <p><i>message</i>—Enters a login message up to 255 characters. You cannot use the delimiting character in the message.</p>

	Command or Action	Purpose
Step 4	end Example: <pre>Device(config)# end</pre>	Returns to privileged EXEC mode.
Step 5	show running-config Example: <pre>Device# show running-config</pre>	Verifies your entries.
Step 6	copy running-config startup-config Example: <pre>Device# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Managing the MAC Address Table

Changing the Address Aging Time

Follow these steps to configure the dynamic address table aging time:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **mac address-table aging-time** [0 | 10-1000000] [**routed-mac** | **vlan** *vlan-id*]
4. **end**
5. **show running-config**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Device> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example:	Enters global configuration mode.

	Command or Action	Purpose
	Device# <code>configure terminal</code>	
Step 3	<p><code>mac address-table aging-time [0 10-1000000]</code> <code>[routed-mac vlan <i>vlan-id</i>]</code></p> <p>Example:</p> <pre>Device(config)# mac address-table aging-time 500 vlan 2</pre>	<p>Sets the length of time that a dynamic entry remains in the MAC address table after the entry is used or updated.</p> <p>The range is 10 to 1000000 seconds. The default is 300. You can also enter 0, which disables aging. Static address entries are never aged or removed from the table.</p> <p><i>vlan-id</i>—Valid IDs are 1 to 4094.</p>
Step 4	<p><code>end</code></p> <p>Example:</p> <pre>Device(config)# end</pre>	Returns to privileged EXEC mode.
Step 5	<p><code>show running-config</code></p> <p>Example:</p> <pre>Device# show running-config</pre>	Verifies your entries.
Step 6	<p><code>copy running-config startup-config</code></p> <p>Example:</p> <pre>Device# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Configuring MAC Address Change Notification Traps

Follow these steps to configure the switch to send MAC address change notification traps to an NMS host:

SUMMARY STEPS

1. `enable`
2. `configure terminal`
3. `snmp-server host host-addr community-string notification-type { informs | traps } { version { 1 | 2c | 3 } } { vrf vrf instance name }`
4. `snmp-server enable traps mac-notification change`
5. `mac address-table notification change`
6. `mac address-table notification change [interval value] [history-size value]`
7. `interface interface-id`
8. `snmp trap mac-notification change { added | removed }`
9. `end`
10. `show running-config`
11. `copy running-config startup-config`

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p>enable</p> <p>Example:</p> <pre>Device> enable</pre>	<p>Enables privileged EXEC mode.</p> <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	<p>configure terminal</p> <p>Example:</p> <pre>Device# configure terminal</pre>	<p>Enters global configuration mode.</p>
Step 3	<p>snmp-server host <i>host-addr</i> <i>community-string</i> <i>notification-type</i> { informs traps } { version { 1 2c 3 } } { vrf <i>vrf instance name</i> }</p> <p>Example:</p> <pre>Device(config)# snmp-server host 172.20.10.10 traps private mac-notification</pre>	<p>Specifies the recipient of the trap message.</p> <ul style="list-style-type: none"> • <i>host-addr</i>—Specifies the name or address of the NMS. • traps (the default)—Sends SNMP traps to the host. • informs—Sends SNMP informs to the host. • version—Specifies the SNMP version to support. Version 1, the default, is not available with informs. • <i>community-string</i>—Specifies the string to send with the notification operation. Though you can set this string by using the snmp-server host command, we recommend that you define this string by using the snmp-server community command before using the snmp-server host command. • <i>notification-type</i>—Uses the mac-notification keyword. • vrf <i>vrf instance name</i>—Specifies the VPN routing/forwarding instance for this host.
Step 4	<p>snmp-server enable traps mac-notification change</p> <p>Example:</p> <pre>Device(config)# snmp-server enable traps mac-notification change</pre>	<p>Enables the device to send MAC address change notification traps to the NMS.</p>
Step 5	<p>mac address-table notification change</p> <p>Example:</p> <pre>Device(config)# mac address-table notification change</pre>	<p>Enables the MAC address change notification feature.</p>

	Command or Action	Purpose
Step 6	<p>mac address-table notification change [<i>interval value</i>] [<i>history-size value</i>]</p> <p>Example:</p> <pre>Device(config)# mac address-table notification change interval 123 Device(config)# mac address-table notification change history-size 100</pre>	<p>Enters the trap interval time and the history table size.</p> <ul style="list-style-type: none"> • (Optional) interval value—Specifies the notification trap interval in seconds between each set of traps that are generated to the NMS. The range is 0 to 2147483647 seconds; the default is 1 second. • (Optional) history-size value—Specifies the maximum number of entries in the MAC notification history table. The range is 0 to 500; the default is 1.
Step 7	<p>interface <i>interface-id</i></p> <p>Example:</p> <pre>Device(config)# interface gigabitethernet 1/0/2</pre>	<p>Enters interface configuration mode, and specifies the Layer 2 interface on which to enable the SNMP MAC address notification trap.</p>
Step 8	<p>snmp trap mac-notification change {added removed}</p> <p>Example:</p> <pre>Device(config-if)# snmp trap mac-notification change added</pre>	<p>Enables the MAC address change notification trap on the interface.</p> <ul style="list-style-type: none"> • Enables the trap when a MAC address is added on this interface. • Enables the trap when a MAC address is removed from this interface.
Step 9	<p>end</p> <p>Example:</p> <pre>Device(config)# end</pre>	<p>Returns to privileged EXEC mode.</p>
Step 10	<p>show running-config</p> <p>Example:</p> <pre>Device# show running-config</pre>	<p>Verifies your entries.</p>
Step 11	<p>copy running-config startup-config</p> <p>Example:</p> <pre>Device# copy running-config startup-config</pre>	<p>(Optional) Saves your entries in the configuration file.</p>

Configuring MAC Address Move Notification Traps

When you configure MAC-move notification, an SNMP notification is generated and sent to the network management system whenever a MAC address moves from one port to another within the same VLAN.

Follow these steps to configure the device to send MAC address-move notification traps to an NMS host:

SUMMARY STEPS

1. enable
2. configure terminal
3. snmp-server host *host-addr* {traps | informs} {version {1 | 2c | 3}} *community-string notification-type*
4. snmp-server enable traps mac-notification move
5. mac address-table notification mac-move
6. end
7. show running-config
8. copy running-config startup-config

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p>enable</p> <p>Example:</p> <pre>Device> enable</pre>	<p>Enables privileged EXEC mode.</p> <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	<p>configure terminal</p> <p>Example:</p> <pre>Device# configure terminal</pre>	<p>Enters global configuration mode.</p>
Step 3	<p>snmp-server host <i>host-addr</i> {traps informs} {version {1 2c 3}} <i>community-string notification-type</i></p> <p>Example:</p> <pre>Device(config)# snmp-server host 172.20.10.10 traps private mac-notification</pre>	<p>Specifies the recipient of the trap message.</p> <ul style="list-style-type: none"> • <i>host-addr</i>—Specifies the name or address of the NMS. • traps (the default)—Sends SNMP traps to the host. • informs—Sends SNMP informs to the host. • version—Specifies the SNMP version to support. Version 1, the default, is not available with informs. • <i>community-string</i>—Specifies the string to send with the notification operation. Though you can set this string by using the snmp-server host command, we recommend that you define this string by using the snmp-server community command before using the snmp-server host command. • <i>notification-type</i>—Uses the mac-notification keyword.
Step 4	<p>snmp-server enable traps mac-notification move</p> <p>Example:</p> <pre>Device(config)# snmp-server enable traps mac-notification move</pre>	<p>Enables the device to send MAC address move notification traps to the NMS.</p>

	Command or Action	Purpose
Step 5	mac address-table notification mac-move Example: <pre>Device(config)# mac address-table notification mac-move</pre>	Enables the MAC address move notification feature.
Step 6	end Example: <pre>Device(config)# end</pre>	Returns to privileged EXEC mode.
Step 7	show running-config Example: <pre>Device# show running-config</pre>	Verifies your entries.
Step 8	copy running-config startup-config Example: <pre>Device# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

What to do next

To disable MAC address-move notification traps, use the **no snmp-server enable traps mac-notification move** global configuration command. To disable the MAC address-move notification feature, use the **no mac address-table notification mac-move** global configuration command.

You can verify your settings by entering the **show mac address-table notification mac-move** privileged EXEC commands.

Configuring MAC Threshold Notification Traps

When you configure MAC threshold notification, an SNMP notification is generated and sent to the network management system when a MAC address table threshold limit is reached or exceeded.

Follow these steps to configure the switch to send MAC address table threshold notification traps to an NMS host:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **snmp-server host *host-addr* {traps / informs} {version {1 | 2c | 3}} *community-string notification-type***
4. **snmp-server enable traps mac-notification threshold**
5. **mac address-table notification threshold**
6. **mac address-table notification threshold [*limit percentage*] | [*interval time*]**

7. `end`
8. `show running-config`
9. `copy running-config startup-config`

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p>enable</p> <p>Example:</p> <pre>Device> enable</pre>	<p>Enables privileged EXEC mode.</p> <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	<p>configure terminal</p> <p>Example:</p> <pre>Device# configure terminal</pre>	<p>Enters global configuration mode.</p>
Step 3	<p>snmp-server host <i>host-addr</i> {traps / informs} {version {1 2c 3}} <i>community-string notification-type</i></p> <p>Example:</p> <pre>Device(config)# snmp-server host 172.20.10.10 traps private mac-notification</pre>	<p>Specifies the recipient of the trap message.</p> <ul style="list-style-type: none"> • <i>host-addr</i>—Specifies the name or address of the NMS. • traps (the default)—Sends SNMP traps to the host. • informs—Sends SNMP informs to the host. • version—Specifies the SNMP version to support. Version 1, the default, is not available with informs. • <i>community-string</i>—Specifies the string to send with the notification operation. You can set this string by using the snmp-server host command, but we recommend that you define this string by using the snmp-server community command before using the snmp-server host command. • <i>notification-type</i>—Uses the mac-notification keyword.
Step 4	<p>snmp-server enable traps mac-notification threshold</p> <p>Example:</p> <pre>Device(config)# snmp-server enable traps mac-notification threshold</pre>	<p>Enables MAC threshold notification traps to the NMS.</p>
Step 5	<p>mac address-table notification threshold</p> <p>Example:</p> <pre>Device(config)# mac address-table notification threshold</pre>	<p>Enables the MAC address threshold notification feature.</p>

	Command or Action	Purpose
Step 6	<p>mac address-table notification threshold [<i>limit percentage</i>] [<i>interval time</i>]</p> <p>Example:</p> <pre>Device(config)# mac address-table notification threshold interval 123 Device(config)# mac address-table notification threshold limit 78</pre>	<p>Enters the threshold value for the MAC address threshold usage monitoring.</p> <ul style="list-style-type: none"> • (Optional) limit percentage—Specifies the percentage of the MAC address table use; valid values are from 1 to 100 percent. The default is 50 percent. • (Optional) interval time—Specifies the time between notifications; valid values are greater than or equal to 120 seconds. The default is 120 seconds.
Step 7	<p>end</p> <p>Example:</p> <pre>Device(config)# end</pre>	<p>Returns to privileged EXEC mode.</p>
Step 8	<p>show running-config</p> <p>Example:</p> <pre>Device# show running-config</pre>	<p>Verifies your entries.</p>
Step 9	<p>copy running-config startup-config</p> <p>Example:</p> <pre>Device# copy running-config startup-config</pre>	<p>(Optional) Saves your entries in the configuration file.</p>

Adding and Removing Static Address Entries

Follow these steps to add a static address:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **mac address-table static mac-addr vlan vlan-id interface interface-id**
4. **end**
5. **show running-config**
6. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p>enable</p> <p>Example:</p>	<p>Enables privileged EXEC mode.</p> <ul style="list-style-type: none"> • Enter your password if prompted.

	Command or Action	Purpose
	Device> <code>enable</code>	
Step 2	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode.
Step 3	mac address-table static <i>mac-addr</i> vlan <i>vlan-id</i> interface <i>interface-id</i> Example: Device(config)# <code>mac address-table static c2f3.220a.12f4 vlan 4 interface gigabitethernet 1/0/1</code>	Adds a static address to the MAC address table. <ul style="list-style-type: none"> • <i>mac-addr</i>—Specifies the destination MAC unicast address to add to the address table. Packets with this destination address received in the specified VLAN are forwarded to the specified interface. • <i>vlan-id</i>—Specifies the VLAN for which the packet with the specified MAC address is received. Valid VLAN IDs are 1 to 4094. • <i>interface-id</i>—Specifies the interface to which the received packet is forwarded. Valid interfaces include physical ports or port channels. For static multicast addresses, you can enter multiple interface IDs. For static unicast addresses, you can enter only one interface at a time, but you can enter the command multiple times with the same MAC address and VLAN ID.
Step 4	end Example: Device(config)# <code>end</code>	Returns to privileged EXEC mode. Alternatively, you can also press Ctrl-Z to exit global configuration mode.
Step 5	show running-config Example: Device# <code>show running-config</code>	Verifies your entries.
Step 6	copy running-config startup-config Example: Device# <code>copy running-config startup-config</code>	(Optional) Saves your entries in the configuration file.

Configuring Unicast MAC Address Filtering

Follow these steps to configure the Device to drop a source or destination unicast static address:

SUMMARY STEPS

1. enable
2. configure terminal
3. mac address-table static *mac-addr* vlan *vlan-id* drop
4. end
5. show running-config
6. copy running-config startup-config

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: <pre>Device> enable</pre>	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: <pre>Device# configure terminal</pre>	Enters global configuration mode.
Step 3	mac address-table static <i>mac-addr</i> vlan <i>vlan-id</i> drop Example: <pre>Device(config)# mac address-table static c2f3.220a.12f4 vlan 4 drop</pre>	Enables unicast MAC address filtering and configure the device to drop a packet with the specified source or destination unicast static address. <ul style="list-style-type: none"> • <i>mac-addr</i>—Specifies a source or destination unicast MAC address (48-bit). Packets with this MAC address are dropped. • <i>vlan-id</i>—Specifies the VLAN for which the packet with the specified MAC address is received. Valid VLAN IDs are 1 to 4094.
Step 4	end Example: <pre>Device(config)# end</pre>	Returns to privileged EXEC mode.
Step 5	show running-config Example: <pre>Device# show running-config</pre>	Verifies your entries.
Step 6	copy running-config startup-config Example:	(Optional) Saves your entries in the configuration file.

Command or Action	Purpose
Device# <code>copy running-config startup-config</code>	

Monitoring and Maintaining Administration of the Device

Command	Purpose
<code>clear mac address-table dynamic</code>	Removes all dynamic entries.
<code>clear mac address-table dynamic address <i>mac-address</i></code>	Removes a specific MAC address.
<code>clear mac address-table dynamic interface <i>interface-id</i></code>	Removes all addresses on the specified physical port or port channel.
<code>clear mac address-table dynamic vlan <i>vlan-id</i></code>	Removes all addresses on a specified VLAN.
<code>show clock [<i>detail</i>]</code>	Displays the time and date configuration.
<code>show ip igmp snooping groups</code>	Displays the Layer 2 multicast entries for all VLANs or the specified VLAN.
<code>show mac address-table address <i>mac-address</i></code>	Displays MAC address table information for the specified MAC address.
<code>show mac address-table aging-time</code>	Displays the aging time in all VLANs or the specified VLAN.
<code>show mac address-table count</code>	Displays the number of addresses present in all VLANs or the specified VLAN.
<code>show mac address-table dynamic</code>	Displays only dynamic MAC address table entries.
<code>show mac address-table interface <i>interface-name</i></code>	Displays the MAC address table information for the specified interface.
<code>show mac address-table move update</code>	Displays the MAC address table move update information.
<code>show mac address-table multicast</code>	Displays a list of multicast MAC addresses.
<code>show mac address-table notification {change mac-move threshold}</code>	Displays the MAC notification parameters and history table.
<code>show mac address-table secure</code>	Displays the secure MAC addresses.
<code>show mac address-table static</code>	Displays only static MAC address table entries.
<code>show mac address-table vlan <i>vlan-id</i></code>	Displays the MAC address table information for the specified VLAN.

Configuration Examples for Device Administration

Example: Setting the System Clock

This example shows how to manually set the system clock:

```
Device# clock set 13:32:00 23 July 2013
```

Examples: Configuring Summer Time

This example (for daylight savings time) shows how to specify that summer time starts on March 10 at 02:00 and ends on November 3 at 02:00:

```
Device(config)# clock summer-time PDT recurring PST date  
10 March 2013 2:00 3 November 2013 2:00
```

This example shows how to set summer time start and end dates:

```
Device(config)#clock summer-time PST date  
20 March 2013 2:00 20 November 2013 2:00
```

Example: Configuring a MOTD Banner

This example shows how to configure a MOTD banner by using the pound sign (#) symbol as the beginning and ending delimiter:

```
Device(config)# banner motd #
```

```
This is a secure site. Only authorized users are allowed.  
For access, contact technical support.
```

```
#
```

```
Device(config)#
```

This example shows the banner that appears from the previous configuration:

```
Unix> telnet 192.0.2.15
```

```
Trying 192.0.2.15...
```

```
Connected to 192.0.2.15.
```

```
Escape character is '^]'.  
#
```

```
This is a secure site. Only authorized users are allowed.
```

```
For access, contact technical support.
```

```
User Access Verification
Password:
```

Example: Configuring a Login Banner

This example shows how to configure a login banner by using the dollar sign (\$) symbol as the beginning and ending delimiter:

```
Device(config)# banner login $
Access for authorized users only. Please enter your username and password.
$
Device(config)#
```

Example: Configuring MAC Address Change Notification Traps

This example shows how to specify 172.20.10.10 as the NMS, enable MAC address notification traps to the NMS, enable the MAC address-change notification feature, set the interval time to 123 seconds, set the history-size to 100 entries, and enable traps whenever a MAC address is added on the specified port:

```
Device(config)# snmp-server host 172.20.10.10 traps private mac-notification
Device(config)# snmp-server enable traps mac-notification change
Device(config)# mac address-table notification change
Device(config)# mac address-table notification change interval 123
Device(config)# mac address-table notification change history-size 100
Device(config)# interface gigabitethernet 1/2/1
Device(config-if)# snmp trap mac-notification change added
```

Example: Configuring MAC Threshold Notification Traps

This example shows how to specify 172.20.10.10 as the NMS, enable the MAC address threshold notification feature, set the interval time to 123 seconds, and set the limit to 78 per cent:

```
Device(config)# snmp-server host 172.20.10.10 traps private mac-notification
Device(config)# snmp-server enable traps mac-notification threshold
Device(config)# mac address-table notification threshold
Device(config)# mac address-table notification threshold interval 123
Device(config)# mac address-table notification threshold limit 78
```

Example: Adding the Static Address to the MAC Address Table

This example shows how to add the static address c2f3.220a.12f4 to the MAC address table. When a packet is received in VLAN 4 with this MAC address as its destination address, the packet is forwarded to the specified port:



Note You cannot associate the same static MAC address to multiple interfaces. If the command is executed again with a different interface, the static MAC address is overwritten on the new interface.

```
Device(config)# mac address-table static c2f3.220a.12f4 vlan 4 interface gigabitethernet
1/1/1
```

Example: Configuring Unicast MAC Address Filtering

This example shows how to enable unicast MAC address filtering and how to configure drop packets that have a source or destination address of c2f3.220a.12f4. When a packet is received in VLAN 4 with this MAC address as its source or destination, the packet is dropped:

```
Device(config)# mac address-table static c2f3.220a.12f4 vlan 4 drop
```

Additional References for Switch Administration

Related Documents

Related Topic	Document Title
Switch administration commands	<i>Catalyst 2960-XR Switch System Management Command Reference</i>
Network management configuration	<i>Catalyst 2960-XR Switch Network Management Configuration Guide</i>
Layer 2 configuration	<i>Catalyst 2960-XR Switch Layer 2 Configuration Guide</i>
VLAN configuration	<i>Catalyst 2960-XR Switch VLAN Management Configuration Guide</i>
Platform-independent command references	<i>Cisco IOS 15.3M&T Command References</i>
Platform-independent configuration information	<i>Cisco IOS 15.3M&T Configuration Guides</i>

Standards and RFCs

Standard/RFC	Title
None	—

MIBs

MIB	MIBs Link
All supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
<p>The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies.</p> <p>To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds.</p> <p>Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.</p>	http://www.cisco.com/support

Feature History and Information for Device Administration

Release	Modification
Cisco IOS Release 15.0(2)EX1	This feature was introduced.



CHAPTER 92

Performing Device Setup Configuration

- [Information About Performing Device Setup Configuration, on page 1847](#)
- [How to Perform Device Setup Configuration, on page 1858](#)
- [Monitoring Device Setup Configuration, on page 1870](#)
- [Configuration Examples for Performing Device Setup, on page 1871](#)
- [Additional References for Performing Switch Setup, on page 1873](#)
- [Feature History and Information For Performing Device Setup Configuration, on page 1874](#)

Information About Performing Device Setup Configuration

Review the sections in this module before performing your initial device configuration tasks that include IP address assignments and DHCP autoconfiguration.

Boot Process

To start your device, you need to follow the procedures in the getting started guide or the hardware installation guide for installing and powering on the device and setting up the initial device configuration (IP address, subnet mask, default gateway, secret and Telnet passwords, and so forth).

The boot loader software performs the normal boot process and includes these activities:

- Locates the bootable (base) package in the bundle or installed package set.
- Performs low-level CPU initialization. It initializes the CPU registers, which control where physical memory is mapped, its quantity, its speed, and so forth.
- Performs power-on self-test (POST) for the CPU subsystem and tests the system DRAM.
- Initializes the file systems on the system board.
- Loads a default operating system software image into memory and boots up the device.

The boot loader provides access to the flash file systems before the operating system is loaded. Normally, the boot loader is used only to load, decompress, and start the operating system. After the boot loader gives the operating system control of the CPU, the boot loader is not active until the next system reset or power-on.

The boot loader also provides trap-door access into the system if the operating system has problems serious enough that it cannot be used. The trap-door operation provides enough access to the system so that if it is

necessary, you can format the flash file system, reinstall the operating system software image by using the Xmodem Protocol, recover from a lost or forgotten password, and finally restart the operating system.

Before you can assign device information, make sure that you have connected a PC or terminal to the console port or a PC to the Ethernet management port, and make sure you have configured the PC or terminal-emulation software baud rate and character format to match that of the device console port settings:

- Baud rate default is 9600.
- Data bits default is 8.



Note If the data bits option is set to 8, set the parity option to none.

- Stop bits default is 2 (minor).
- Parity settings default is none.

Devices Information Assignment

You can assign IP information through the device setup program, through a DHCP server, or manually.

Use the device setup program if you want to be prompted for specific IP information. With this program, you can also configure a hostname and an enable secret password.

It gives you the option of assigning a Telnet password (to provide security during remote management) and configuring your switch as a command or member switch of a cluster or as a standalone switch.

Use a DHCP server for centralized control and automatic assignment of IP information after the server is configured.



Note If you are using DHCP, do not respond to any of the questions in the setup program until the device receives the dynamically assigned IP address and reads the configuration file.

If you are an experienced user familiar with the device configuration steps, manually configure the device. Otherwise, use the setup program described in the *Boot Process* section.

Default Switch Information

Table 178: Default Switch Information

Feature	Default Setting
IP address and subnet mask	No IP address or subnet mask are defined.
Default gateway	No default gateway is defined.
Enable secret password	No password is defined.
Hostname	The factory-assigned default hostname is Device.

Feature	Default Setting
Telnet password	No password is defined.
Cluster command switch functionality	Disabled.
Cluster name	No cluster name is defined.

DHCP-Based Autoconfiguration Overview

DHCP provides configuration information to Internet hosts and internetworking devices. This protocol consists of two components: one for delivering configuration parameters from a DHCP server to a device and an operation for allocating network addresses to devices. DHCP is built on a client-server model, in which designated DHCP servers allocate network addresses and deliver configuration parameters to dynamically configured devices. The device can act as both a DHCP client and a DHCP server.

During DHCP-based autoconfiguration, your device (DHCP client) is automatically configured at startup with IP address information and a configuration file.

With DHCP-based autoconfiguration, no DHCP client-side configuration is needed on your device. However, you need to configure the DHCP server for various lease options associated with IP addresses.

If you want to use DHCP to relay the configuration file location on the network, you might also need to configure a Trivial File Transfer Protocol (TFTP) server and a Domain Name System (DNS) server.



Note We recommend a redundant connection between a switch stack and the DHCP, DNS, and TFTP servers. This is to help ensure that these servers remain accessible in case one of the connected stack members is removed from the switch stack.

The DHCP server for your device can be on the same LAN or on a different LAN than the device. If the DHCP server is running on a different LAN, you should configure a DHCP relay device between your device and the DHCP server. A relay device forwards broadcast traffic between two directly connected LANs. A router does not forward broadcast packets, but it forwards packets based on the destination IP address in the received packet.

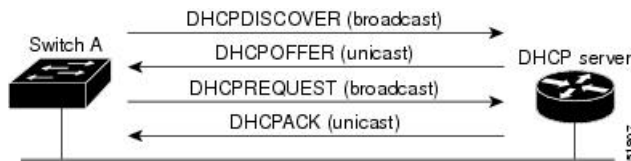
DHCP-based autoconfiguration replaces the BOOTP client functionality on your device.

DHCP Client Request Process

When you boot up your device, the DHCP client is invoked and requests configuration information from a DHCP server when the configuration file is not present on the device. If the configuration file is present and the configuration includes the **ip address dhcp** interface configuration command on specific routed interfaces, the DHCP client is invoked and requests the IP address information for those interfaces.

This is the sequence of messages that are exchanged between the DHCP client and the DHCP server.

Figure 139: DHCP Client and Server Message Exchange



The client, Device A, broadcasts a DHCPDISCOVER message to locate a DHCP server. The DHCP server offers configuration parameters (such as an IP address, subnet mask, gateway IP address, DNS IP address, a lease for the IP address, and so forth) to the client in a DHCPOFFER unicast message.

In a DHCPREQUEST broadcast message, the client returns a formal request for the offered configuration information to the DHCP server. The formal request is broadcast so that all other DHCP servers that received the DHCPDISCOVER broadcast message from the client can reclaim the IP addresses that they offered to the client.

The DHCP server confirms that the IP address has been allocated to the client by returning a DHCPACK unicast message to the client. With this message, the client and server are bound, and the client uses configuration information received from the server. The amount of information the device receives depends on how you configure the DHCP server.

If the configuration parameters sent to the client in the DHCPOFFER unicast message are invalid (a configuration error exists), the client returns a DHCPDECLINE broadcast message to the DHCP server.

The DHCP server sends the client a DHCPNAK denial broadcast message, which means that the offered configuration parameters have not been assigned, that an error has occurred during the negotiation of the parameters, or that the client has been slow in responding to the DHCPOFFER message (the DHCP server assigned the parameters to another client).

A DHCP client might receive offers from multiple DHCP or BOOTP servers and can accept any of the offers; however, the client usually accepts the first offer it receives. The offer from the DHCP server is not a guarantee that the IP address is allocated to the client; however, the server usually reserves the address until the client has had a chance to formally request the address. If the device accepts replies from a BOOTP server and configures itself, the device broadcasts, instead of unicasts, TFTP requests to obtain the device configuration file.

The DHCP hostname option allows a group of devices to obtain hostnames and a standard configuration from the central management DHCP server. A client (device) includes in its DHCPDISCOVER message an option 12 field used to request a hostname and other configuration parameters from the DHCP server. The configuration files on all clients are identical except for their DHCP-obtained hostnames.

If a client has a default hostname (the **hostname name** global configuration command is not configured or the **no hostname** global configuration command is entered to remove the hostname), the DHCP hostname option is not included in the packet when you enter the **ip address dhcp** interface configuration command. In this case, if the client receives the DHCP hostname option from the DHCP interaction while acquiring an IP address for an interface, the client accepts the DHCP hostname option and sets the flag to show that the system now has a hostname configured.

DHCP-based Autoconfiguration and Image Update

You can use the DHCP image upgrade features to configure a DHCP server to download both a new image and a new configuration file to one or more devices in a network. Simultaneous image and configuration upgrade for all switches in the network helps ensure that each new device added to a network receives the same image and configuration.

There are two types of DHCP image upgrades: DHCP autoconfiguration and DHCP auto-image update.

Restrictions for DHCP-based Autoconfiguration

- The DHCP-based autoconfiguration with a saved configuration process stops if there is not at least one Layer 3 interface in an up state without an assigned IP address in the network.
- Unless you configure a timeout, the DHCP-based autoconfiguration with a saved configuration feature tries indefinitely to download an IP address.
- The auto-install process stops if a configuration file cannot be downloaded or if the configuration file is corrupted.
- The configuration file that is downloaded from TFTP is merged with the existing configuration in the running configuration but is not saved in the NVRAM unless you enter the **write memory** or **copy running-configuration startup-configuration** privileged EXEC command. If the downloaded configuration is saved to the startup configuration, the feature is not triggered during subsequent system restarts.

DHCP Autoconfiguration

DHCP autoconfiguration downloads a configuration file to one or more devices in your network from a DHCP server. The downloaded configuration file becomes the running configuration of the device. It does not overwrite the bootup configuration saved in the flash, until you reload the device.

DHCP Auto-Image Update

You can use DHCP auto-image upgrade with DHCP autoconfiguration to download both a configuration and a new image to one or more devices in your network. The device (or devices) downloading the new configuration and the new image can be blank (or only have a default factory configuration loaded).

If the new configuration is downloaded to a switch that already has a configuration, the downloaded configuration is appended to the configuration file stored on the switch. (Any existing configuration is not overwritten by the downloaded one.)

To enable a DHCP auto-image update on the device, the TFTP server where the image and configuration files are located must be configured with the correct option 67 (the configuration filename), option 66 (the DHCP server hostname) option 150 (the TFTP server address), and option 125 (description of the Cisco IOS image file) settings.

After you install the device in your network, the auto-image update feature starts. The downloaded configuration file is saved in the running configuration of the device, and the new image is downloaded and installed on the device. When you reboot the device, the configuration is stored in the saved configuration on the device.

DHCP Server Configuration Guidelines

Follow these guidelines if you are configuring a device as a DHCP server:

- You should configure the DHCP server with reserved leases that are bound to each device by the device hardware address.
- If you want the device to receive IP address information, you must configure the DHCP server with these lease options:
 - IP address of the client (required)

- Subnet mask of the client (required)
- DNS server IP address (optional)
- Router IP address (default gateway address to be used by the device) (required)
- If you want the device to receive the configuration file from a TFTP server, you must configure the DHCP server with these lease options:
 - TFTP server name (required)
 - Boot filename (the name of the configuration file that the client needs) (recommended)
 - Hostname (optional)
- Depending on the settings of the DHCP server, the device can receive IP address information, the configuration file, or both.
- If you do not configure the DHCP server with the lease options described previously, it replies to client requests with only those parameters that are configured. If the IP address and the subnet mask are not in the reply, the device is not configured. If the router IP address or the TFTP server name are not found, the device might send broadcast, instead of unicast, TFTP requests. Unavailability of other lease options does not affect autoconfiguration.
- The device can act as a DHCP server. By default, the Cisco IOS DHCP server and relay agent features are enabled on your device but are not configured. (These features are not operational.)

Purpose of the TFTP Server

Based on the DHCP server configuration, the device attempts to download one or more configuration files from the TFTP server. If you configured the DHCP server to respond to the device with all the options required for IP connectivity to the TFTP server, and if you configured the DHCP server with a TFTP server name, address, and configuration filename, the device attempts to download the specified configuration file from the specified TFTP server.

If you did not specify the configuration filename, the TFTP server, or if the configuration file could not be downloaded, the device attempts to download a configuration file by using various combinations of filenames and TFTP server addresses. The files include the specified configuration filename (if any) and these files: `network-config`, `cisconet.cfg`, `hostname.config`, or `hostname.cfg`, where *hostname* is the device's current hostname. The TFTP server addresses used include the specified TFTP server address (if any) and the broadcast address (255.255.255.255).

For the device to successfully download a configuration file, the TFTP server must contain one or more configuration files in its base directory. The files can include these files:

- The configuration file named in the DHCP reply (the actual device configuration file).
- The `network-config` or the `cisconet.cfg` file (known as the default configuration files).
- The `router-config` or the `ciscottr.cfg` file (These files contain commands common to all devices. Normally, if the DHCP and TFTP servers are properly configured, these files are not accessed.)

If you specify the TFTP server name in the DHCP server-lease database, you must also configure the TFTP server name-to-IP-address mapping in the DNS-server database.

If the TFTP server to be used is on a different LAN from the device, or if it is to be accessed by the device through the broadcast address (which occurs if the DHCP server response does not contain all the required information described previously), a relay must be configured to forward the TFTP packets to the TFTP server. The preferred solution is to configure the DHCP server with all the required information.

Purpose of the DNS Server

The DHCP server uses the DNS server to resolve the TFTP server name to an IP address. You must configure the TFTP server name-to-IP address map on the DNS server. The TFTP server contains the configuration files for the device.

You can configure the IP addresses of the DNS servers in the lease database of the DHCP server from where the DHCP replies will retrieve them. You can enter up to two DNS server IP addresses in the lease database.

The DNS server can be on the same LAN or on a different LAN from the device. If it is on a different LAN, the device must be able to access it through a router.

How to Obtain Configuration Files

Depending on the availability of the IP address and the configuration filename in the DHCP reserved lease, the device obtains its configuration information in these ways:

- The IP address and the configuration filename is reserved for the device and provided in the DHCP reply (one-file read method).

The device receives its IP address, subnet mask, TFTP server address, and the configuration filename from the DHCP server. The device sends a unicast message to the TFTP server to retrieve the named configuration file from the base directory of the server and upon receipt, it completes its boot up process.

- The IP address and the configuration filename is reserved for the device, but the TFTP server address is not provided in the DHCP reply (one-file read method).

The device receives its IP address, subnet mask, and the configuration filename from the DHCP server. The device sends a broadcast message to a TFTP server to retrieve the named configuration file from the base directory of the server, and upon receipt, it completes its boot-up process.

- Only the IP address is reserved for the device and provided in the DHCP reply. The configuration filename is not provided (two-file read method).

The device receives its IP address, subnet mask, and the TFTP server address from the DHCP server. The device sends a unicast message to the TFTP server to retrieve the `network-config` or `cisconet.cfg` default configuration file. (If the `network-config` file cannot be read, the device reads the `cisconet.cfg` file.)

The default configuration file contains the hostnames-to-IP-address mapping for the device. The device fills its host table with the information in the file and obtains its hostname. If the hostname is not found in the file, the device uses the hostname in the DHCP reply. If the hostname is not specified in the DHCP reply, the device uses the default *Switch* as its hostname.

After obtaining its hostname from the default configuration file or the DHCP reply, the device reads the configuration file that has the same name as its hostname (`hostname-config` or `hostname.cfg`, depending on whether `network-config` or `cisconet.cfg` was read earlier) from the TFTP server. If the `cisconet.cfg` file is read, the filename of the host is truncated to eight characters.

If the device cannot read the `network-config`, `cisconet.cfg`, or the hostname file, it reads the `router-config` file. If the device cannot read the `router-config` file, it reads the `ciscortr.cfg` file.



Note The device broadcasts TFTP server requests if the TFTP server is not obtained from the DHCP replies, if all attempts to read the configuration file through unicast transmissions fail, or if the TFTP server name cannot be resolved to an IP address.

How to Control Environment Variables

With a normally operating device, you enter the boot loader mode only through the console connection. Unplug the switch power cord, then reconnect the power cord. Hold down the **MODE** button until you see the boot loader switch prompt

The device boot loader software provides support for nonvolatile environment variables, which can be used to control how the boot loader or any other software running on the system, functions. Boot loader environment variables are similar to environment variables that can be set on UNIX or DOS systems.

Environment variables that have values are stored in flash memory outside of the flash file system.

Each line in these files contains an environment variable name and an equal sign followed by the value of the variable. A variable has no value if it is not present; it has a value if it is listed even if the value is a null string. A variable that is set to a null string (for example, “”) is a variable with a value. Many environment variables are predefined and have default values.

Environment variables store two kinds of data:

- Data that controls code, which does not read the Cisco IOS configuration file. For example, the name of a boot loader helper file, which extends or patches the functionality of the boot loader can be stored as an environment variable.
- Data that controls code, which is responsible for reading the Cisco IOS configuration file. For example, the name of the Cisco IOS configuration file can be stored as an environment variable.

You can change the settings of the environment variables by accessing the boot loader or by using Cisco IOS commands. Under normal circumstances, it is not necessary to alter the setting of the environment variables.

Common Environment Variables

This table describes the function of the most common environment variables.

Table 179: Common Environment Variables

Variable	Boot Loader Command	Cisco IOS Global Configuration Command
BOOT	<p>set BOOT <i>filesystem</i> <i>:/file-url ...</i></p> <p>A semicolon-separated list of executable files to try to load and execute when automatically booting. If the BOOT environment variable is not set, the system attempts to load and execute the first executable image it can find by using a recursive, depth-first search through the flash file system. If the BOOT variable is set but the specified images cannot be loaded, the system attempts to boot the first bootable file that it can find in the flash file system.</p>	<p>boot system <i>{filesystem : /file-url ...</i></p> <p>Specifies the Cisco IOS image to load during the next boot cycle and the stack members on which the image is loaded. This command changes the setting of the BOOT environment variable.</p>
MANUAL_BOOT	<p>set MANUAL_BOOT yes</p> <p>Decides whether the switch automatically or manually boots.</p> <p>Valid values are 1, yes, 0, and no. If it is set to no or 0, the boot loader attempts to automatically boot up the system. If it is set to anything else, you must manually boot up the switch from the boot loader mode.</p>	<p>boot manual</p> <p>Enables manually booting the switch during the next boot cycle and changes the setting of the MANUAL_BOOT environment variable.</p> <p>The next time you reboot the system, the switch is in boot loader mode. To boot up the system, use the boot flash: <i>filesystem :/file-url</i> boot loader command, and specify the name of the bootable image.</p>

Variable	Boot Loader Command	Cisco IOS Global Configuration Command
CONFIG_FILE	<p>set CONFIG_FILE flash:/ file-url</p> <p>Changes the filename that Cisco IOS uses to read and write a nonvolatile copy of the system configuration.</p>	<p>boot config-file flash:/ file-url</p> <p>Specifies the filename that Cisco IOS uses to read and write a nonvolatile copy of the system configuration. This command changes the CONFIG_FILE environment variable.</p>
SWITCH_NUMBER	<p>set SWITCH_NUMBER stack-member-number</p> <p>Changes the member number of a stack member.</p>	<p>switch current-stack-member-number renumber new-stack-member-number</p> <p>Changes the member number of a stack member.</p>
SWITCH_PRIORITY	<p>set SWITCH_PRIORITY stack-member-number</p> <p>Changes the priority value of a stack member.</p>	<p>switch stack-member-number priority priority-number</p> <p>Changes the priority value of a stack member.</p>
BAUD	<p>set BAUD baud-rate</p>	<p>line console 0 speed speed-value</p> <p>Configures the baud rate.</p>
ENABLE_BREAK	<p>set ENABLE_BREAK yes/no</p>	<p>boot enable-break switch yes/no</p> <p>This command can be issued when the flash filesystem is initialized when ENABLE_BREAK is set to yes.</p>

Environment Variables for TFTP

When the switch is connected to a PC through the Ethernet management port, you can download or upload a configuration file to the boot loader by using TFTP. Make sure the environment variables in this table are configured.

Table 180: Environment Variables for TFTP

Variable	Description
MAC_ADDR	Specifies the MAC address of the switch. Note We recommend that you do not modify this variable. However, if you modify this variable after the boot loader is up or the value is different from the saved value, enter this command before using TFTP. A reset is required for the new value to take effect.
IP_ADDRESS	Specifies the IP address and the subnet mask for the associated IP subnet of the switch.
DEFAULT_ROUTER	Specifies the IP address and subnet mask of the default gateway.

Scheduled Reload of the Software Image

You can schedule a reload of the software image to occur on the device at a later time (for example, late at night or during the weekend when the device is used less), or you can synchronize a reload network-wide (for example, to perform a software upgrade on all devices in the network).



Note A scheduled reload must take place within approximately 24 days.

You have these reload options:

- Reload of the software to take effect in the specified minutes or hours and minutes. The reload must take place within approximately 24 hours. You can specify the reason for the reload in a string up to 255 characters in length.
- Reload of the software to take place at the specified time (using a 24-hour clock). If you specify the month and day, the reload is scheduled to take place at the specified time and date. If you do not specify the month and day, the reload takes place at the specified time on the current day (if the specified time is later than the current time) or on the next day (if the specified time is earlier than the current time). Specifying 00:00 schedules the reload for midnight.

The **reload** command halts the system. If the system is not set to manually boot up, it reboots itself.

If your device is configured for manual booting, do not reload it from a virtual terminal. This restriction prevents the device from entering the boot loader mode and then taking it from the remote user's control.

If you modify your configuration file, the device prompts you to save the configuration before reloading. During the save operation, the system requests whether you want to proceed with the save if the CONFIG_FILE environment variable points to a startup configuration file that no longer exists. If you proceed in this situation, the system enters setup mode upon reload.

To cancel a previously scheduled reload, use the **reload cancel** privileged EXEC command.

How to Perform Device Setup Configuration

Using DHCP to download a new image and a new configuration to a device requires that you configure at least two devices. One device acts as a DHCP and TFTP server and the second device (client) is configured to download either a new configuration file or a new configuration file and a new image file.

Configuring DHCP Autoconfiguration (Only Configuration File)

This task describes how to configure DHCP autoconfiguration of the TFTP and DHCP settings on an existing device in the network so that it can support the autoconfiguration of a new device.

SUMMARY STEPS

1. **configure terminal**
2. **ip dhcp pool** *poolname*
3. **boot** *filename*
4. **network** *network-number mask prefix-length*
5. **default-router** *address*
6. **option 150** *address*
7. **exit**
8. **tftp-server flash:***filename.text*
9. **interface** *interface-id*
10. **no switchport**
11. **ip address** *address mask*
12. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 2	ip dhcp pool <i>poolname</i> Example: Device(config)# ip dhcp pool pool	Creates a name for the DHCP server address pool, and enters DHCP pool configuration mode.
Step 3	boot <i>filename</i> Example:	Specifies the name of the configuration file that is used as a boot image.

	Command or Action	Purpose
	Device (dhcp-config) # <code>boot config-boot.text</code>	
Step 4	<p>network <i>network-number mask prefix-length</i></p> <p>Example:</p> <pre>Device (dhcp-config) # network 10.10.10.0 255.255.255.0</pre>	<p>Specifies the subnet network number and mask of the DHCP address pool.</p> <p>Note The prefix length specifies the number of bits that comprise the address prefix. The prefix is an alternative way of specifying the network mask of the client. The prefix length must be preceded by a forward slash (/).</p>
Step 5	<p>default-router <i>address</i></p> <p>Example:</p> <pre>Device (dhcp-config) # default-router 10.10.10.1</pre>	Specifies the IP address of the default router for a DHCP client.
Step 6	<p>option 150 <i>address</i></p> <p>Example:</p> <pre>Device (dhcp-config) # option 150 10.10.10.1</pre>	Specifies the IP address of the TFTP server.
Step 7	<p>exit</p> <p>Example:</p> <pre>Device (dhcp-config) # exit</pre>	Returns to global configuration mode.
Step 8	<p>tftp-server flash:<i>filename.text</i></p> <p>Example:</p> <pre>Device (config) # tftp-server flash:config-boot.text</pre>	Specifies the configuration file on the TFTP server.
Step 9	<p>interface <i>interface-id</i></p> <p>Example:</p> <pre>Device (config) # interface gigabitethernet 1/0/4</pre>	Specifies the address of the client that will receive the configuration file.
Step 10	<p>no switchport</p> <p>Example:</p> <pre>Device (config-if) # no switchport</pre>	Puts the interface into Layer 3 mode.

	Command or Action	Purpose
Step 11	ip address <i>address mask</i> Example: <pre>Device(config-if)# ip address 10.10.10.1 255.255.255.0</pre>	Specifies the IP address and mask for the interface.
Step 12	end Example: <pre>Device(config-if)# end</pre>	Returns to privileged EXEC mode.

Configuring DHCP Auto-Image Update (Configuration File and Image)

This task describes DHCP autoconfiguration to configure TFTP and DHCP settings on an existing device to support the installation of a new switch.

Before you begin

You must first create a text file (for example, `autoinstall_dhcp`) that will be uploaded to the device. In the text file, put the name of the image that you want to download (forexample, `c3750e-ipservices-mz.122-44.3.SE.tarc3750x-ipservices-mz.122-53.3.SE2.tar`). This image must be a tar and not a bin file.

SUMMARY STEPS

1. **configure terminal**
2. **ip dhcp pool** *poolname*
3. **boot** *filename*
4. **network** *network-number mask prefix-length*
5. **default-router** *address*
6. **option 150** *address*
7. **option 125** *hex*
8. **copy tftp flash** *filename.txt*
9. **copy tftp flash** *imagename.bin*
10. **exit**
11. **tftp-server flash:** *config.text*
12. **tftp-server flash:** *imagename.bin*
13. **tftp-server flash:** *filename.txt*
14. **interface** *interface-id*
15. **no switchport**
16. **ip address** *address mask*
17. **end**
18. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 2	ip dhcp pool <i>poolname</i> Example: Device(config)# ip dhcp pool pool1	Creates a name for the DHCP server address pool and enter DHCP pool configuration mode.
Step 3	boot <i>filename</i> Example: Device(dhcp-config)# boot config-boot.text	Specifies the name of the file that is used as a boot image.
Step 4	network <i>network-number mask prefix-length</i> Example: Device(dhcp-config)# network 10.10.10.0 255.255.255.0	Specifies the subnet network number and mask of the DHCP address pool. Note The prefix length specifies the number of bits that comprise the address prefix. The prefix is an alternative way of specifying the network mask of the client. The prefix length must be preceded by a forward slash (/).
Step 5	default-router <i>address</i> Example: Device(dhcp-config)# default-router 10.10.10.1	Specifies the IP address of the default router for a DHCP client.
Step 6	option 150 <i>address</i> Example: Device(dhcp-config)# option 150 10.10.10.1	Specifies the IP address of the TFTP server.
Step 7	option 125 <i>hex</i> Example: Device(dhcp-config)# option 125 hex 0000.0009.0a05.08661.7574.6f69.6e73.7461.6c6c.5f64.686370	Specifies the path to the text file that describes the path to the image file.

	Command or Action	Purpose
Step 8	copy tftp flash <i>filename.txt</i> Example: Device(config)# copy tftp flash image.bin	Uploads the text file to the device.
Step 9	copy tftp flash <i>imagename.bin</i> Example: Device(config)# copy tftp flash image.bin	Uploads the tar file for the new image to the device.
Step 10	exit Example: Device(dhcp-config)# exit	Returns to global configuration mode.
Step 11	tftp-server flash: <i>config.text</i> Example: Device(config)# tftp-server flash:config-boot.text	Specifies the Cisco IOS configuration file on the TFTP server.
Step 12	tftp-server flash: <i>imagename.bin</i> Example: Device(config)# tftp-server flash:image.bin	Specifies the image name on the TFTP server.
Step 13	tftp-server flash: <i>filename.txt</i> Example: Device(config)# tftp-server flash:boot-config.text	Specifies the text file that contains the name of the image file to download
Step 14	interface <i>interface-id</i> Example: Device(config)# interface gigabitethernet 1/0/4	Specifies the address of the client that will receive the configuration file.
Step 15	no switchport Example: Device(config-if)# no switchport	Puts the interface into Layer 3 mode.

	Command or Action	Purpose
Step 16	ip address <i>address mask</i> Example: <pre>Device(config-if)# ip address 10.10.10.1 255.255.255.0</pre>	Specifies the IP address and mask for the interface.
Step 17	end Example: <pre>Device(config-if)# end</pre>	Returns to privileged EXEC mode.
Step 18	copy running-config startup-config Example: <pre>Device(config-if)# end</pre>	(Optional) Saves your entries in the configuration file.

Configuring the Client to Download Files from DHCP Server



Note You should only configure and enable the Layer 3 interface. Do not assign an IP address or DHCP-based autoconfiguration with a saved configuration.

SUMMARY STEPS

1. **configure terminal**
2. **boot host dhcp**
3. **boot host retry timeout** *timeout-value*
4. **banner config-save** ^C *warning-message* ^C
5. **end**
6. **show boot**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>Device# configure terminal</pre>	Enters global configuration mode.
Step 2	boot host dhcp Example:	Enables autoconfiguration with a saved configuration.

	Command or Action	Purpose
	<code>Device(conf)# boot host dhcp</code>	
Step 3	<p>boot host retry timeout <i>timeout-value</i></p> <p>Example:</p> <pre>Device(conf)# boot host retry timeout 300</pre>	<p>(Optional) Sets the amount of time the system tries to download a configuration file.</p> <p>Note If you do not set a timeout, the system will try indefinitely to obtain an IP address from the DHCP server.</p>
Step 4	<p>banner config-save ^C <i>warning-message</i> ^C</p> <p>Example:</p> <pre>Device(conf)# banner config-save ^C Caution - Saving Configuration File to NVRAM May Cause You to No longer Automatically Download Configuration Files at Reboot^C</pre>	(Optional) Creates warning messages to be displayed when you try to save the configuration file to NVRAM.
Step 5	<p>end</p> <p>Example:</p> <pre>Device(config-if)# end</pre>	Returns to privileged EXEC mode.
Step 6	<p>show boot</p> <p>Example:</p> <pre>Device# show boot</pre>	Verifies the configuration.

Manually Assigning IP Information to Multiple SVIs

This task describes how to manually assign IP information to multiple switched virtual interfaces (SVIs):



Note If the switch is running the IP Lite image, you can also manually assign IP information to a port if you first put the port into Layer 3 mode by using the **no switchport** interface configuration command.

SUMMARY STEPS

1. **configure terminal**
2. **interface vlan** *vlan-id*
3. **ip address** *ip-address subnet-mask*
4. **exit**
5. **ip default-gateway** *ip-address*
6. **end**

7. `show interfaces vlan vlan-id`
8. `show ip redirects`

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode.
Step 2	interface vlan <i>vlan-id</i> Example: Device(config)# <code>interface vlan 99</code>	Enters interface configuration mode, and enter the VLAN to which the IP information is assigned. The range is 1 to 4094.
Step 3	ip address <i>ip-address subnet-mask</i> Example: Device(config-vlan)# <code>ip address 10.10.10.2 255.255.255.0</code>	Enters the IP address and subnet mask.
Step 4	exit Example: Device(config-vlan)# <code>exit</code>	Returns to global configuration mode.
Step 5	ip default-gateway <i>ip-address</i> Example: Device(config)# <code>ip default-gateway 10.10.10.1</code>	<p>Enters the IP address of the next-hop router interface that is directly connected to the device where a default gateway is being configured. The default gateway receives IP packets with unresolved destination IP addresses from the device.</p> <p>Once the default gateway is configured, the device has connectivity to the remote networks with which a host needs to communicate.</p> <p>Note When your device is configured to route with IP, it does not need to have a default gateway set.</p>
Step 6	end Example: Device(config)# <code>end</code>	Returns to privileged EXEC mode.

	Command or Action	Purpose
Step 7	show interfaces vlan <i>vlan-id</i> Example: Device# show interfaces vlan 99	Verifies the configured IP address.
Step 8	show ip redirects Example: Device# show ip redirects	Verifies the configured default gateway.

Configuring the NVRAM Buffer Size

The default NVRAM buffer size is 512 KB. In some cases, the configuration file might be too large to save to NVRAM. Typically, this occurs when you have many switches in a switch stack. You can configure the size of the NVRAM buffer to support larger configuration files. The new NVRAM buffer size is synced to all current and new member switches.



Note After you configure the NVRAM buffer size, reload the switch or switch stack.

When you add a switch to a stack and the NVRAM size differs, the new switch syncs with the stack and reloads automatically.

SUMMARY STEPS

1. **configure terminal**
2. **boot buffersize** *size*
3. **end**
4. **show boot**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 2	boot buffersize <i>size</i> Example: Device(config)# boot buffersize 524288	Configures the NVRAM buffersize in KB. The valid range for <i>size</i> is from 4096 to 1048576.

	Command or Action	Purpose
Step 3	end Example: <pre>Device(config)# end</pre>	Returns to privileged EXEC mode.
Step 4	show boot Example: <pre>Device# show boot</pre>	Verifies the configuration.

Modifying the Device Startup Configuration

Specifying the Filename to Read and Write the System Configuration

By default, the Cisco IOS software uses the `config.text` file to read and write a nonvolatile copy of the system configuration. However, you can specify a different filename, which will be loaded during the next boot cycle.

Before you begin

Use a standalone device for this task.

SUMMARY STEPS

1. **configure terminal**
2. **boot flash:/file-url**
3. **end**
4. **show boot**
5. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>Switch# configure terminal</pre>	Enters global configuration mode.
Step 2	boot flash:/file-url Example: <pre>Switch(config)# boot flash:config.text</pre>	Specifies the configuration file to load during the next boot cycle. <i>file-url</i> —The path (directory) and the configuration filename. Filenames and directory names are case-sensitive.

	Command or Action	Purpose
Step 3	end Example: Switch(config)# end	Returns to privileged EXEC mode.
Step 4	show boot Example: Switch# show boot	Verifies your entries. The boot global configuration command changes the setting of the CONFIG_FILE environment variable.
Step 5	copy running-config startup-config Example: Switch# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Manually Booting the Switch

By default, the switch automatically boots up; however, you can configure it to manually boot up.

Before you begin

Use a standalone switch for this task.

SUMMARY STEPS

1. **configure terminal**
2. **boot manual**
3. **end**
4. **show boot**
5. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 2	boot manual Example: Device(config)# boot manual	Enables the switch to manually boot up during the next boot cycle.

	Command or Action	Purpose
Step 3	end Example: <pre>Device(config)# end</pre>	Returns to privileged EXEC mode.
Step 4	show boot Example: <pre>Device# show boot</pre>	<p>Verifies your entries.</p> <p>The boot manual global command changes the setting of the MANUAL_BOOT environment variable.</p> <p>The next time you reboot the system, the switch is in boot loader mode, shown by the <i>switch:</i> prompt. To boot up the system, use the boot filesystem:/file-url boot loader command.</p> <ul style="list-style-type: none"> • <i>filesystem:</i>—Uses flash: for the system board flash device. <pre>Switch: boot flash:</pre> • For <i>file-url</i>—Specifies the path (directory) and the name of the bootable image. <p>Filenames and directory names are case-sensitive.</p>
Step 5	copy running-config startup-config Example: <pre>Device# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Configuring a Scheduled Software Image Reload

This task describes how to configure your device to reload the software image at a later time.

SUMMARY STEPS

1. **configure terminal**
2. **copy running-config startup-config**
3. **reload in [hh:]mm [text]**
4. **reload at hh: mm [month day | day month] [text]**
5. **reload cancel**
6. **show reload**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example:	Enters global configuration mode.

	Command or Action	Purpose
	Device# <code>configure terminal</code>	
Step 2	copy running-config startup-config Example: <code>copy running-config startup-config</code>	Saves your device configuration information to the startup configuration before you use the reload command.
Step 3	reload in [hh:]mm [text] Example: Device(config)# <code>reload in 12</code> System configuration has been modified. Save? [yes/no]: <code>y</code>	Schedules a reload of the software to take affect in the specified minutes or hours and minutes. The reload must take place within approximately 24 days. You can specify the reason for the reload in a string up to 255 characters in length.
Step 4	reload at hh: mm [month day day month] [text] Example: Device(config)# <code>reload at 14:00</code>	Specifies the time in hours and minutes for the reload to occur. Note Use the at keyword only if the device system clock has been set (through Network Time Protocol (NTP), the hardware calendar, or manually). The time is relative to the configured time zone on the device. To schedule reloads across several devices to occur simultaneously, the time on each device must be synchronized with NTP.
Step 5	reload cancel Example: Device(config)# <code>reload cancel</code>	Cancels a previously scheduled reload.
Step 6	show reload Example: <code>show reload</code>	Displays information about a previously scheduled reload or identifies if a reload has been scheduled on the device.

Monitoring Device Setup Configuration

Example: Verifying the Device Running Configuration

```
Device# show running-config
Building configuration...

Current configuration: 1363 bytes
!
```



```

version 12.4
no service pad
service timestamps debug uptime
service timestamps log uptime
no service password-encryption
!
hostname Stack1
!
enable secret 5 $1$ej9.$DMUvAUnZOAmvmgqBEzIxEO
!
.
<output truncated>
.
interface gigabitethernet6/0/2
mvr type source

<output truncated>

...!
interface VLAN1
 ip address 172.20.137.50 255.255.255.0
 no ip directed-broadcast
!
ip default-gateway 172.20.137.1 !
!
snmp-server community private RW
snmp-server community public RO
snmp-server community private@es0 RW
snmp-server community public@es0 RO
snmp-server chassis-id 0x12
!
end

```

Examples: Displaying Software Install

This example displays software bootup in install mode:

```
switch# boot flash:/c2960x-universalk9-mz-150-2.EX1/c2960x-universalk9-mz-150-2.EX1.bin
```

Configuration Examples for Performing Device Setup

Example: Configuring a Device as a DHCP Server

```

Device# configure terminal
Device(config)# ip dhcp pool pool1
Device(dhcp-config)# network 10.10.10.0 255.255.255.0
Device(dhcp-config)# boot config-boot.text
Device(dhcp-config)# default-router 10.10.10.1
Device(dhcp-config)# option 150 10.10.10.1
Device(dhcp-config)# exit
Device(config)# tftp-server flash:config-boot.text
Device(config)# interface gigabitethernet 1/0/4
Device(config-if)# no switchport
Device(config-if)# ip address 10.10.10.1 255.255.255.0

```

```
Device(config-if)# end
```

Example: Configuring DHCP Auto-Image Update

```
Device# configure terminal
Device(config)# ip dhcp pool pool1
Device(dhcp-config)# network 10.10.10.0 255.255.255.0
Device(dhcp-config)# boot config-boot.text
Device(dhcp-config)# default-router 10.10.10.1
Device(dhcp-config)# option 150 10.10.10.1
Device(dhcp-config)# option 125 hex 0000.0009.0a05.08661.7574.6f69.6e73.7461.6c6c.5f64.686370

Device(dhcp-config)# exit
Device(config)# tftp-server flash:config-boot.text
Device(config)# tftp-server flash:image_name
Device(config)# tftp-server flash:boot-config.text
Device(config)# tftp-server flash:autoinstall_dhcp
Device(config)# interface gigabitethernet 1/0/4
Device(config-if)# no switchport
Device(config-if)# ip address 10.10.10.1 255.255.255.0
Device(config-if)# end
```

Example: Configuring a Device to Download Configurations from a DHCP Server

This example uses a Layer 3 SVI interface on VLAN 99 to enable DHCP-based autoconfiguration with a saved configuration:

```
Device# configure terminal
Device(config)# boot host dhcp
Device(config)# boot host retry timeout 300
Device(config)# banner config-save ^C Caution - Saving Configuration File to NVRAM May Cause
  You to No longer Automatically Download Configuration Files at Reboot^C
Device(config)# vlan 99
Device(config-vlan)# interface vlan 99
Device(config-if)# no shutdown
Device(config-if)# end
Device# show boot
BOOT path-list:
Config file:          flash:/config.text
Private Config file: flash:/private-config.text
Enable Break:        no
Manual Boot:         no
HELPER path-list:
NVRAM/Config file
  buffer size:       32768
Timeout for Config
  Download:          300 seconds
Config Download
  via DHCP:         enabled (next boot: enabled)
Device#
```

Example: Configuring NVRAM Buffer Size

```

Device# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Device(config)# boot buffersize 600000
Device(config)# end
Device# show boot
BOOT path-list      :
Config file         : flash:/config.text
Private Config file : flash:/private-config.text
Enable Break       : no
Manual Boot        : no
HELPER path-list   :
Auto upgrade       : yes
Auto upgrade path  :
NVRAM/Config file
  buffer size:     600000
Timeout for Config
  Download:        300 seconds
Config Download
  via DHCP:        enabled (next boot: enabled)
Device#

```

Additional References for Performing Switch Setup

Related Documents

Related Topic	Document Title
Switch setup commands Boot loader commands	<i>Catalyst 2960-XR Switch System Management Command Reference</i>
USB flash devices	<i>Catalyst 2960-XR Switch Interface and Hardware Component Configuration Guide</i> <i>Catalyst 2960-XR Switch Managing Cisco IOS Image Files Configuration Guide</i>
Hardware installation	<i>Catalyst 2960-XR Switch Hardware Installation Guide</i>
Platform-independent command references	<i>Cisco IOS 15.3M&T Command References</i>
Platform-independent configuration information	<i>Cisco IOS 15.3M&T Configuration Guides</i>

Standards and RFCs

Standard/RFC	Title
None	—

MIBs

MIB	MIBs Link
All supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
<p>The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies.</p> <p>To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds.</p> <p>Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.</p>	http://www.cisco.com/support

Feature History and Information For Performing Device Setup Configuration

Command History	Release	Modification
	Cisco IOS Release 15.0(2)EX1	This feature was introduced.



CHAPTER 93

Configuring AVC with DNS-AS

- [Prerequisites for AVC with DNS-AS, on page 1875](#)
- [Restrictions and Guidelines for AVC with DNS-AS, on page 1875](#)
- [Information About AVC with DNS-AS, on page 1876](#)
- [How to Configure AVC with DNS-AS, on page 1880](#)
- [Monitoring AVC with DNS-AS, on page 1893](#)
- [Troubleshooting AVC with DNS-AS, on page 1896](#)
- [Feature History and Information for AVC with DNS-AS, on page 1897](#)

Prerequisites for AVC with DNS-AS

- You have the [Cisco ONE for Access](#) license to use AVC with DNS-AS.
- You have enabled Multilayer Switch (MLS) Quality of Service (QoS).
- You have maintained metadata in the authoritative DNS server and reachability exists - before you enable AVC with DNS-AS
- The DNS-AS client can snoop forward look-up requests originating from hosts.
- To ensure DNS packet logging or snooping, you have attached the policy map to the interface, by using the **service-policy input** command.

Restrictions and Guidelines for AVC with DNS-AS

- Only a forward look-up is supported.
- Two DNS servers are supported (in case of a failover). One is considered the primary DNS server and other, the secondary DNS server.
- IPv6 is not supported—AAAA requests, and IPv6 DNS servers are not supported.
- AVC with DNS-AS is supported only on physical interfaces, in the ingress direction.
- Virtual Routing and Forwarding (VRF) is not supported.

- We recommend a maximum of 300 AVC with DNS-AS applications (domain names) in the binding table, because of its effect on the ternary content addressable memory (TCAM). To know how the addition of applications affects the TCAM see the [Troubleshooting AVC with DNS-AS](#) section of this chapter.

Information About AVC with DNS-AS

The Application Visibility Control (AVC) with Domain Name System as an Authoritative Source (DNS-AS) feature (AVC with DNS-AS) provides a centralized means of controlling the identification and classification of trusted network traffic in an organization. It accomplishes this by using network metadata stored in a DNS server that is authoritative to the domain in question, to identify applications, Quality of Service (QoS) to classify the corresponding traffic and apply suitable policies, and Flexible NetFlow (FNF), to monitor and export application information to an external collector.

The feature provides:

- **Application Visibility**—Ensuring unambiguous visibility of applications.

The DNS-AS mechanism snoops requests and does not require a CPU-intensive, deep packet inspection (DPI). Since traffic classification is by means of a DNS request and not DPI, this feature is compatible in scenarios where network traffic is encrypted.

- **Metadata Driven**—Using information about applications.

You can program the network holistically so it behaves like a self-driving car. You now have information about all the required applications in your network, irrespective of whether traffic is encrypted or not.

- **Centralized Control**—Using a cross-domain application intent policy controller.

The feature leverages an existing, universally available query-response mechanism to enable local DNS servers within an organization to act as authoritative servers and propagate application classification information to DNS-AS clients (switches) in an enterprise network.

- **Control without Administrative Access**—Providing alternatives to controller-based approaches.

The feature supports scenarios where your network may be in the cloud and you may not own it. You can still control network devices across the Internet, even though you may not have administrative control of these devices.

Overview of AVC with DNS-AS

The process starts with an organization's requirements relating to management and control of network traffic. You begin by assessing the software applications that run on the various hosts (phones, PCs etc.) in your network, the domains (websites) and applications accessed by these devices, and the business-relevance of these domains and applications in your organization.

The assessment helps you arrive at a list of domains and applications that are "trusted" by your organization, designating all remaining domains and applications as untrusted.

With DNS-AS enabled on your network and the list of trusted domains at hand, the networking devices or DNS-AS clients in your network identify which applications the network traffic belongs to or which domains are being requested. As long as the traffic is part of the trusted list, the switch requests the DNS server for metadata and IP address information. This request is sent in the form of a DNS-query. The response, once

received, is cached locally until the Time-to-Live (TTL) for that resource record expires. The response is bound to the traffic and allows the DNS-AS client to now identify, classify, and forward traffic accordingly.

Key Concepts for AVC with DNS-AS

Concept	Meaning or Definition
Metadata (RFC6759)	<p>In the context of the AVC with DNS-AS feature, this includes traffic classification information, application identification information, and business relevance information.</p> <p>Metadata is maintained in the form of TXT records. The following is a sample metadata record in the prescribed format:</p> <pre>CISCO-CLS=app-name:example app-class:TD business:YES app-id:CU/28202</pre>
Forward look-up	<p>A request for an IP address or a request for an “A” record, originating from a host.</p> <p>Being able to snoop these forward lookups in the network traffic is fundamental to the AVC with DNS-AS feature.</p>
Host	<p>A PC or mobile where users run software applications, access websites and so on.</p> <p>Forward look-up requests originate from hosts.</p>
Client or DNS-AS client	<p>Networking devices throughout your network. Host traffic is always routed through such a client.</p> <p>Note This chapter deals with the configuration of the AVC with DNS-AS on Cisco Catalyst Switches that are deployed as access switches only. Throughout this document, the term client, DNS-AS client, refers to the switch where AVC with DNS-AS is enabled.</p> <p>DNS-AS clients receive metadata from an authoritative DNS server and maintain a database of this information in the form of records. How long the record remains in the client’s database, is determined by the record’s TTL.</p>
Binding table	<p>A table that resides in the DNS-AS client and serves as a database of parsed DNS server responses [TXT records and “A” records].</p> <p>Every DNS-AS client has a binding table of its own.</p> <p>This table not to be confused with the trusted domain list which is only a list of the trusted domains.</p>
"A" record	<p>A record containing the domain name and IP address information [Only IPv4 address]. This is one of the DNS-Server responses (the other being the TXT record) and has a predefined lifespan.</p> <p>A forward lookup request from a host is a request for an “A” record.</p>

Concept	Meaning or Definition
TXT DNS-AS resource record or TXT record	<p>A record containing metadata. This is one of the DNS-Server responses (the other being the “A” record) and has a predefined lifespan.</p> <p>A TXT record is limited to 255 characters.</p> <p>For AVC with DNS-AS, the TXT attribute is always CISCO-CLS. Any TXT record that starts with CISCO-CLS= can be recognized as an AVC with DNS-AS message. The message format is as follows:</p> <pre>CISCO-CLS=<option>:<val>{ <option>:<val>}*</pre>
Time-to-Live (TTL)	<p>The lifespan of an “A” record and TXT record in the binding table.</p> <p>TTL values are configured on the DNS server.</p> <p>While a TTL accompanies both TXT and “A” record responses, the DNS client only goes by the “A” record response from the DNS server.</p>
Authoritative DNS server	<p>The go-to DNS server for all client metadata and “A” record requests.</p> <p>Every DNS domain has only one authoritative DNS server.</p> <p>Such a server maintains records of application metadata in the form of a TXT record, and only returns responses to queries about domain names that have been maintained in the required format.</p> <p>The following is a sample metadata record in the prescribed format:</p> <pre>CISCO-CLS=app-name:example app-class:TD business:YES app-id:CU/28202</pre>

AVC with DNS-AS Process Flow

The working of AVC with DNS-AS involves the DNS snooping process and the DNS-AS client process—both of which are loosely coupled, but independent processes.

DNS Snooping Process

-
- Step 1** The host initiates an “A” record request.
- A user from your organization is in a meeting room in an office building. The associated DNS-AS client here is a switch (Network traffic from this meeting room is routed through this switch). The user looks up a website `www.example.com`, which initiates the request for an “A” record.
- Step 2** The authoritative DNS-server responds with an “A” record response.
-

DNS-AS Client Process

-
- Step 1** The DNS-AS client sends a DNS query (TXT request) to the authoritative DNS server.

The DNS-AS client, which is constantly snooping for requests (that correspond with entries in the trusted domain list), finds the host's forward look-up request. Based on the snooped result, the DNS-AS client sends a TXT request to the authoritative DNS server.

Note The DNS-AS client receives a copy of the host's "A" record request, and does not alter the host's original request in any manner.

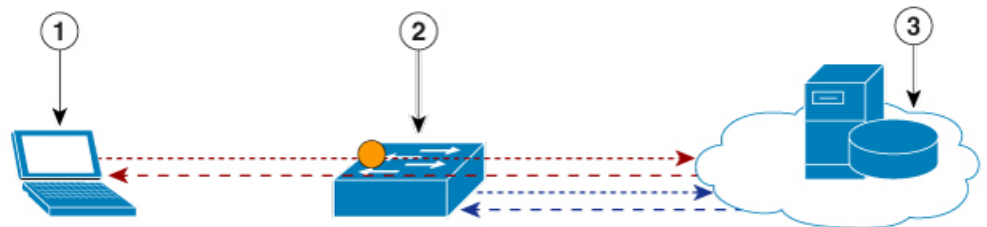
- Step 2** The authoritative DNS-server responds with a TXT record response.
- Step 3** A successful TXT response is followed by an "A" record request.
- Step 4** The authoritative DNS-server responds with an "A" record response.
- Step 5** The DNS-AS client parses and saves the response in its binding table.

The DNS-AS client saves the TXT record and "A" record in its binding table. The response will remain saved in the binding table for the duration specified by the TTL of the "A" record. The system automatically checks and prevents duplicate entries for a fully qualified domain name in the binding table.

The DNS-AS client uses the metadata it receives (from the DNS Server), to determine if a QoS policy should be applied.

The DNS-AS client forwards information about identified applications, to FNF, enabling you to export this information.

Figure: AVC with DNS-AS Process Flow



1	Host	2	DNS-As Client	3	Authoritative DNS Server
Part I: DNS Snooping Process					
	An "A" record request from the host to the DNS server		An "A" record response from the DNS server to the host		
Part II: DNS-As Client Process					
	A copy of the host's "A" record request that the DNS-AS client saves	-	-		
	TXT record and "A" record request from the DNS-AS client to the DNS server		TXT record and "A" record response from the DNS server to the DNS-AS client		

Stacking and AVC with DNS-AS

AVC with DNS-AS supports stacking. To ensure successful stack management, the binding table database of the active stack (also a DNS-AS client) is synchronized with the stack member (also a DNS-AS client). As long as AVC with DNS-AS is enabled, no additional user configuration is required. The binding table entries are synchronized at these times:

- The stack member comes up (bulk synchronization).
- New entries are added to the binding table database.
- One or more entries are cleared from the database.

Default Configuration for AVC with DNS-AS

DNS-AS is disabled.

How to Configure AVC with DNS-AS

Generating Metadata Streams

Application metadata is configured and saved on the local, authoritative DNS server. You configure application classification information, for each trusted domain, in a prescribed format (a metadata stream). This is the information that the server propagates to switches when queried for application metadata. When the switch sends a TXT query regarding an application, the DNS server sends the relevant metadata in the TXT response.

To generate metadata streams, perform the following task:

SUMMARY STEPS

1. Go to the: [AVC Resource Record Generator](#).
2. Click one of the options to generate the metadata stream.
 - Generate predefined
 - Generate custom
3. Copy metadata into the corresponding TXT Resource Record of the DNS server in charge of the DNS domain that you have marked as a trusted domain.

DETAILED STEPS

	Command or Action	Purpose
Step 1	Go to the: AVC Resource Record Generator . Example: <code>CISCO-CLS=app-name:example app-class:TD business:YES app-id:CU/28202</code>	Helps you generate a metadata stream for an application or domain, in a TXT record format. You can specify the following metadata fields: <ul style="list-style-type: none"> • (Optional) Domain Name

	Command or Action	Purpose
		<ul style="list-style-type: none"> • (Mandatory) Application Name—A value is mandatory. This can be an existing application name or custom application name. <ul style="list-style-type: none"> • Existing Application Name (app-name:)—Select from the list of standard applications. • (Optional) Custom Application Name(app-name:)—If you enter a custom application name, you must also maintain the Traffic Class and Business Relevance information in the metadata stream. • (Optional) Selector ID (app-id:)—Consists of a classification engine ID (first eight bits) and a selector ID (the next twenty-four bits). <ul style="list-style-type: none"> • Engine ID or Classification Engine ID—Defines the context for the selector ID. Only these engine IDs are allowed: <ul style="list-style-type: none"> L3—IANA layer 3 protocol number L4—IANA layer 4 well-known port number L7—Cisco global application ID CU—Custom protocol. Use this engine ID for custom application names. • Selector ID—An application identifier, for a given classification engine ID. Enter a numeric value between 1 and 65535 <p>Note When you enter the engine ID and selector ID for existing application names, be sure to align with the Network Based Application Recognition (NBAR) standard. Only then will the FNF exporters report with a common ID and in a consistent manner.</p> • (Optional) Port Range (server-port:) • (Optional) Traffic Class (app-class:) • (Optional) Business Relevance (business:)—If you do not select yes or no, the business relevance value is set based on the app-class or app-name, in that order of priority.

	Command or Action	Purpose
		For information about how traffic class and business relevance fields here map to QoS traffic classification, see App-Class and QoS Traffic Mapping
Step 2	<p>Click one of the options to generate the metadata stream.</p> <ul style="list-style-type: none"> • Generate predefined • Generate custom <p>Example: Generate predefined</p>	<p>Generate predefined—Generate a predefined metadata stream for well known applications, using best practice defaults.</p> <p>Generate custom—Generate a custom metadata stream for your own applications using custom values.</p>
Step 3	Copy metadata into the corresponding TXT Resource Record of the DNS server in charge of the DNS domain that you have marked as a trusted domain.	Copy and paste the metadata stream from the website, to the authoritative DNS server you are using.

Configuring a DNS Server as the Authoritative Server

All DNS-AS clients in the network should be configured to send all DNS queries to one authoritative DNS server. On a Cisco Catalyst switch, perform the following task:

SUMMARY STEPS

1. **configure terminal**
2. **ip name-server***server-address*

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p>configure terminal</p> <p>Example:</p> <pre>Device# configure terminal</pre>	Enters global configuration mode.
Step 2	<p>ip name-server<i>server-address</i></p> <p>Example:</p> <pre>Device(config)# ip name-server server-address 192.0.2.1 192.0.2.2</pre>	<p>Specifies the address of the authoritative DNS server. The port number is always 53.</p> <p>You can configure up to two DNS Servers, in case of a failover.</p>

	Command or Action	Purpose
		<p>Note The command allows you configure up to six name servers (IPv4 and IPv6). Ensure that at least the first two IP addresses in the sequence are IPv4 addresses, because the AVC with DNS-AS feature will use only these. See the example below, here the first two addresses are IPv4 (192.0.2.1 and 192.0.2.2), the third one (2001:DB8::1) is an IPv6 address. AVC with DNS-AS will use the first two.</p> <pre>Device(config)# ip name-server 192.0.2.1 192.0.2.2 2001:DB8::1</pre>

Enabling AVC with DNS-AS

DNS-AS is disabled by default. To enable the feature on a Cisco Catalyst switch, perform the following task:

SUMMARY STEPS

1. **configure terminal**
2. **[no] avc dns-as client enable**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: <pre>Device# configure terminal</pre>	Enters global configuration mode.
Step 2	[no] avc dns-as client enable Example: <pre>Device(config)# avc dns-as client enable</pre>	Enables AVC with DNS-AS on the switch (DNS-AS client). The system then creates a binding table where parsed DNS server responses are stored till the TTL expires. <p>Note To ensure DNS packet logging or snooping, you must attach the policy map (containing the relevant class maps that will determine traffic class) to the interface by using the service-policy input command. For more information, see Configuring QoS for AVC with DNS-AS, on page 1884</p>

Maintaining the List of Trusted Domains

Trusted domains are saved in every DNS-AS client where AVC with DNS-AS is enabled. When the feature is first enabled on the DNS-AS client, the list is empty. You must enter the domains that the switch should

trust. The switch snoops only for network traffic that is maintained in this list. To make entries in the trusted domain list, perform the following task:

SUMMARY STEPS

1. **configure terminal**
2. **[no] avc dns-as client trusted-domains**
3. **[no] domain domain-name**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 2	[no] avc dns-as client trusted-domains Example: Device(config)# avc dns-as client trusted-domains	Enters the trusted domain configuration mode.
Step 3	[no] domain domain-name Example: Device(config-trusted-domains)# domain www.example.com OR Device(config-trusted-domains)# domain *example.com	Enter the domain name you would like to add to the trusted domain list. This forms part of the list of trusted domains for the DNS-AS client. All remaining domains are ignored and will follow default forwarding behavior. You can enter up to 50 domains. You can use regular expressions to match the domain name. For example, to represent all the domains for an organization, if you enter:Switch(config-trusted-domains)# domain *.example.*, the DNS-AS client matches www.example.com, ftp.example.org and any other domain that pertains to the organization “example”. But use such an entry at your discretion, because it could increase the size of the binding table considerably.

Configuring QoS for AVC with DNS-AS

In order to isolate and classify trusted traffic as defined in the metadata stream, you must create class maps (one for each traffic class) > define traffic-class match criteria and business-relevance match criteria > create a policy map > add the class map > set action > attach the policy map to the interface. For more information, see the [Classification Overview](#) section of the *Configuring QoS* chapter in this guide.

Class Map Configuration in the Easy QoS Model

In order to determine the number of traffic classes that should be provisioned, you can use the 12-class Easy QoS Model. This model provides a uniform, standards-based recommendations to help ensure

that QoS designs and deployments are unified and consistent across an organization. The following sample output displays class map configuration for traffic class and business relevance, according to the 12-class Easy QoS Model:



Note Only in the context of the DNS-AS feature, you can specify up to two match attributes for each class.

```
class-map match-all VOICE
match protocol attribute traffic-class voip-telephony
match protocol attribute business-relevance business-relevant
class-map match-all BROADCAST-VIDEO
match protocol attribute traffic-class broadcast-video
match protocol attribute business-relevance business-relevant
class-map match-all REAL-TIME-INTERACTIVE
match protocol attribute traffic-class real-time-interactive
match protocol attribute business-relevance business-relevant
class-map match-all MULTIMEDIA-CONFERENCING
match protocol attribute traffic-class multimedia-conferencing
match protocol attribute business-relevance business-relevant
class-map match-all MULTIMEDIA-STREAMING
match protocol attribute traffic-class multimedia-streaming
match protocol attribute business-relevance business-relevant
class-map match-all SIGNALING
match protocol attribute traffic-class signaling
match protocol attribute business-relevance business-relevant
class-map match-all NETWORK-CONTROL
match protocol attribute traffic-class network-control
match protocol attribute business-relevance business-relevant
class-map match-all NETWORK-MANAGEMENT
match protocol attribute traffic-class ops-admin-mgmt
match protocol attribute business-relevance business-relevant
class-map match-all TRANSACTIONAL-DATA
match protocol attribute traffic-class transactional-data
match protocol attribute business-relevance business-relevant
class-map match-all BULK-DATA
match protocol attribute traffic-class bulk-data
match protocol attribute business-relevance business-relevant
class-map match-all SCAVENGER
match protocol attribute business-relevance business-irrelevant
```

Policy Map Definitions in the Easy QoS Model

The following sample output displays the policy map definitions, with traffic attribute marking for all the traffic classes in the 12-class Easy QoS Model:

```
policy-map MARKING
class VOICE
set dscp ef
class BROADCAST-VIDEO
set dscp cs5
class REAL-TIME-INTERACTIVE
set dscp cs4
class MULTIMEDIA-CONFERENCING
set dscp af41
class MULTIMEDIA-STREAMING
set dscp af31
```

```

class SIGNALING
set dscp cs3
class NETWORK-CONTROL
set dscp cs6
class NETWORK-MANAGEMENT
set dscp cs2
class TRANSACTIONAL-DATA
set dscp af21
class BULK-DATA
set dscp af11
class SCAVENGER
set dscp cs1
class class-default
set dscp default

```

App-Class and QoS Traffic Mapping

The following table shows how the `app-class` field in the metadata stream maps to the 12-class Easy QoS Model of traffic classification.

App-Class and QoS Traffic Mapping

Application Class Long Text	Application Class Short Text	Corresponding QoS Traffic Class Name and Business Relevance
VOIP-TELEPHONY	VO	Traffic-class = voip-telephony Business-relevance = YES
BROADCAST-VIDEO	BV	Traffic-class = broadcast-video Business-relevance = YES
REALTIME-INTERACTIVE	RTI	Traffic-class = real-time-interactive Business-relevance = YES
MULTIMEDIA-CONFERENCING	MMC	Traffic-class = multimedia-conferencing Business-relevance = YES
MULTIMEDIA-STREAMING	MMS	Traffic-class = multimedia-streaming Business-relevance = YES
NETWORK-CONTROL	NC	Traffic-class = network-control Business-relevance = YES
SIGNALING	CS	Traffic-class = Signaling Business-relevance = YES
OPS-ADMIN-MGMT	OAM	Traffic-class = ops-admin-mgmt Business-relevance = YES

Application Class Long Text	Application Class Short Text	Corresponding QoS Traffic Class Name and Business Relevance
TRANSACTIONAL-DATA	TD	Traffic-class = Transactional-Data Business-relevance = YES
BULK-DATA	BD	Traffic-class = bulk-data Business-relevance = YES
BEST-EFFORT	BE	Traffic-class = <no change> Business-relevance = default
SCAVENGER	SCV	Traffic-Class = <no change> Business-relevance = NO

Classifying Network Control Traffic

The following example shows how to classify network control traffic. The corresponding metadata that should be maintained is: CISCO-CLS=app-name:example|app-class:NC|business:YES

1. Create class maps and match attributes:

```
Device# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Device(config)# class-map NETWORK-CONTROL
Device(config-cmap)# match protocol attribute traffic-class network-control
Device(config-cmap)# match protocol attribute business-relevance business-relevant
Device(config-cmap)# end
```

2. Create the policy map, attach the class map to it and specify priority:

```
Device# configure terminal
Device configuration commands, one per line. End with CNTL/Z.
Device(config)# policy-map MARKING
Device(config-pmap)# class NETWORK-CONTROL
Device(config-pmap-c)# set dscp ef
Device(config-pmap-c)# end
```

3. Attach the policy map to an interface:

```
Device# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Device(config)# interface tengigabitethernet 1/0/1
Device(config-if)# service-policy input MARKING
Device(config-if)# end
```

Configuring FNF for AVC with DNS-AS

With FNF you can gain visibility into the applications running on your network, and use FNF option templates to export application ID, description, and attribute information. You must configure these FNF settings on the DNS-AS client:

- Configure a flow record to collect nonkey field **application-name**, and the key fields **ipv4 source address** and **ipv4 destination address**

- Configure a flow exporter and the two option templates. Option templates fetch application information.

Option template **application-table**, exports only applications resolved by the DNS-AS client, that is, the application ID and name from the binding table. The corresponding application descriptions are from Network Based Application Recognition (NBAR) definition for standard applications. A constructed help string is used for custom applications..

Option template **application-attributes** fetches attribute information by mapping it to the application name. Where standard application names are used, the option template uses standard Network Based Application Recognition (NBAR) attribute definitions; where custom application names are used, user-defined application names and only certain attribute fields are guaranteed to carry values.

- Configure a flow monitor and apply it to an interface to enable network traffic monitoring.

FNF Interaction with DNS-AS—With every flow that is created in the flow table, the DNS-AS client resolves the application name for the flow (if the entry exists in the binding table), by using the destination IP address (and if not available), the source IP address.

At periodic, configured intervals (600 seconds, by default), FNF exports option template data, that is mapped to the corresponding application name, to an external collector.

Option Templates

The **application-table** and **application-attributes** option templates are supported. Option templates determine the information that is exported to an external collector.

option application-table

This template exports the application name, application tag, and description to the external collector.

On a device where AVC with DNS-AS is enabled, only applications resolved by the DNS-AS client are exported. But as a permanent feature, the application-table template exports applications **unclassified** and **unknown**, irrespective of whether the feature is enabled or not.

- Application Name—For custom and standard applications, this information is derived from the TXT response (**app-name:**) that is saved in the binding table.
- Application Tag—The same as the application ID in the context of the AVC with DNS-AS feature. It consists of the engine ID and selector ID.
 - Engine ID or Classification Engine ID—Defines the context for the selector ID. Only these values are supported:
 - L3—IANA layer 3 protocol number (IANA_L3_STANDARD, ID: 1)
 - L4—IANA layer 4 well-known port number (IANA_L4_STANDARD, ID: 3)
 - L7—Cisco global application ID (CISCO_L7_GLOBAL, ID: 13)
 - CU—Custom protocol, (NBAR_CUSTOM, ID: 6)
 - Selector ID—Uniquely identifies the application or classification.

For standard applications, the application tag information is derived from these sources, in the given order of precedence:

1. TXT response (**app-id:**)

2. The NBAR definition for standard applications (if the TXT response does not carry a value).

For custom applications, the following applies to application tag information:

- It is derived only from the TXT response (**app-id:**)
 - For the engine ID, the DNS-AS client automatically uses CU—Custom protocol, (NBAR_CUSTOM, ID: 6).
 - For the selector ID, the DNS-AS client allots a custom selector ID. A maximum of 120 custom applications are supported - out of which 110 are available to the DNS-AS client. Starting with selector ID value 243, IDs are assigned in descending order. When there are no remaining IDs to assign, the entry is not saved in the binding table.
- Description—This information is derived from the NBAR definition for standard applications. For custom applications, the DNS-AS client uses: User Defined Protocol <app-name>.

option application-attributes

This template enables the collector to map the application names (from the option application-table), to attributes. Attributes are statically assigned to each protocol or application, and are not dependent on traffic. The template supports the following attributes:

For standard applications—

- Application Tag—See the Application Tag info in the [option application-table](#) section above. The same applies here.
- Category—Groups applications based on the first level of categorization for each protocol as the match criteria. Similar applications are grouped together under one category. For example, the email category contains all email applications such as, Internet Mail Access Protocol (IMAP), Simple Mail Transfer Protocol (SMTP), Lotus Notes, and so on.
- Sub-category—Groups applications based on the second level of categorization for each protocol as the match criteria. For example, clearcase, dbase, rda, mysql and other database applications are grouped under the database group.
- Application Group—Groups the same networking applications together. For instance, Example-Messenger, Example-VoIP-messenger, and Example-VoIP-over-SIP are grouped together under the example-messenger-group
- Peer-to-peer (p2p)—Groups protocols based on whether or not they use p2p technology.
- Tunnel—Groups protocols based on whether or not a protocol tunnels the traffic of other protocols. Protocols for which the NBAR does not provide any value are categorized under the unassigned tunnel group. For example, Layer 2 Tunneling Protocols (L2TP).
- Encryption—Groups applications based on the encrypted and nonencrypted status of the applications. Protocols for which the NBAR does not provide any value are categorized under the unassigned encrypted group.
- Traffic class—Groups applications and protocols based on the traffic class they belong to. For example, all applications that have traffic class TD. Traffic class information is derived from these sources, in the given order of precedence:

1. TXT response (**app-class:**)

2. The NBAR definition for standard applications (if the TXT response does not carry a value)
- Business relevance—Groups applications based on whether or not they have been marked as business-relevant. For example, all applications that have business relevance as YES. Business relevance information is derived from these sources, in the given order of precedence:
 1. TXT response (**business:**)
 2. The NBAR definition for standard applications (if the TXT response does not carry a value)

For custom applications—

Only these attributes of the application-attributes options template are guaranteed to carry a value:

- Application Tag—See the Application Tag info in the [option application-table](#) section above. The same applies here.
- Traffic class—This information is derived from the TXT response (**app-class:**)
- Business relevance—This information is derived from the TXT response (**business:**)

Sample FNF Configuration for AVC with DNS-AS

The following example shows how you can configure FNF for AVC with DNS-AS:

Part 1: Create a flow record. As in the example, you must configure:

- The source and destination IP addresses as key fields, in order to resolve application names.
- The use of the application name as a monkey field in flow record.

Additionally (not mandatory), you can also configure the number of bytes or packets in a flow as a monkey field, to display the number of applications sent to the collector

```
Device# configure terminal
Device(config)# flow record example-record1
Device(config-flow-record)# match ipv4 source address
Device(config-flow-record)# match ipv4 destination address
Device(config-flow-record)# collect application name
Device(config-flow-record)# collect counter packets
Device(config-flow-record)# exit
```

```
Device# show flow record example-record1
flow record example-record1
  match ipv4 source address
  match ipv4 destination address
  collect application name
  collect counter packets
```

Part 2: Create a flow exporter.

Also configure the **application-table** and **application-attributes** option templates in the exporter. Without option templates, the collector cannot retrieve meaningful application information. At a minimum we recommend that you configure the application-table option. For attribute information, also configure the application-attribute option.

You can also change the frequency of template export in seconds (the allowed range is 1 to 86400 seconds; the default is 600 seconds).

```

Device(config)# flow exporter example-exporter1
Device(config-flow-exporter)# option application-table
Device(config-flow-exporter)# option application-attributes
Device(config-flow-exporter)# template data timeout 500
Device(config-flow-exporter)# exit

Device# show flow exporter example-exporter1
Flow Exporter example-exporter1:
  Description:                User defined
  Export protocol:            NetFlow Version 9
  Transport Configuration:
    Destination IP address: 192.0.1.254
    Source IP address:      192.51.100.2
    Transport Protocol:    UDP
    Destination Port:      9995
    Source Port:           54964
    DSCP:                   0x0
    TTL:                    255
    Output Features:       Not Used
  Options Configuration:
    application-table (timeout 500 seconds)
    application-attributes (timeout 500 seconds)

Device# show flow exporter example-exporter1 statistics
Flow Exporter example-exporter1:
  Packet send statistics (last cleared 00:00:48 ago):
    Successfully sent:      2                (924 bytes)

  Client send statistics:
    Client: Option options application-name
      Records added:        4
      - sent:               4
      Bytes added:         332
      - sent:               332

    Client: Option options application-attributes
      Records added:        2
      - sent:               2
      Bytes added:         388
      - sent:               388

```

Part 3: Create a flow monitor

Apply the flow monitor to an interface, to perform network traffic monitoring.

You can also apply a QoS policy to the same interface. This example applies the QoS policy created as part of the sample QoS configuration ([Classifying Network Control Traffic, on page 1887](#))

```

Device# configure terminal
Device(config)# flow monitor example-monitor1
Device(config-flow-monitor)# record example-record1
Device(config-flow-monitor)# exporter exporter-exporter1
Device(config-flow-monitor)# exit
Device(config)# interface tengigabitethernet 1/0/1
Device(config-if)# switchport access vlan 100
Device(config-if)# switchport mode access
Device(config-if)# ip flow monitor example-monitor1 input
Device(config-if)# service-policy input MARKING
Device(config-if)# end

Device# show flow monitor
flow monitor example-monitor1
  record example-record1

```

```

exporter example-exporter1
!
Device# show interface tengigabitethernet1/0/1
interface tengigabitethernet1/0/1
  switchport access vlan 100
  switchport mode access
ip flow monitor example-monitor1 input

Device# show flow monitor example-monitor1 cache
Cache type:                Normal
Cache size:                16640
Current entries:          3
High Watermark:           3

Flows added:              6
Flows aged:               3
  - Active timeout        ( 1800 secs) 0
  - Inactive timeout      (   30 secs) 3
  - Event aged            0
  - Watermark aged        0
  - Emergency aged        0

IPV4 SOURCE ADDRESS:      192.0.1.254
IPV4 DESTINATION ADDRESS: 192.51.100.2
counter packets long:     7479
application name:         appexample1

IPV4 SOURCE ADDRESS:      192.51.100.11
IPV4 DESTINATION ADDRESS: 203.0.113.125
counter packets long:     445
application name:         appexample2

IPV4 SOURCE ADDRESS:      192.51.51.51
IPV4 DESTINATION ADDRESS: 203.0.113.100
counter packets long:     14325
application name:         appexample3
Switch#

```

Part 4: Other related show commands

```

Device# show avc dns-as client binding-table detail
DNS-AS generated protocols:
Max number of protocols      :50
Customization interval [min] :N/A

Age          : The amount of time that the entry is active
TTL          : Time to live which was learned from DNS-AS server
Time To Expire : Entry expiration time in case device does not see DNS traffic for the
entry host

Protocol-Name      : appexample1
VRF                : <default>
Host               : www.appexample1.com
Age[min]           : 2
TTL[min]           : 60
Time To Expire[min] : 58
TXT Record         : app-name:appexample1|app-class:VO|business:YES
Traffic Class      : voip-telephony
Business Relevance : business relevant
IP                 : 192.0.1.254

Protocol-Name      : appexample2
VRF                : <default>

```

```

Host                : www.appexample2.com
Age[min]            : 2
TTL[min]           : 60
Time To Expire[min] : 58
TXT Record         : app-name:appexample2|app-class:VO|business:YES
Traffic Class      : voip-telephony
Business Relevance : business relevant
IP                 : 192.51.100.11

```

<output truncated>

Device# **show flow exporter option application engines**

```

Engine: prot (IANA_L3_STANDARD, ID: 1)
Engine: port (IANA_L4_STANDARD, ID: 3)
Engine: NBAR (NBAR_CUSTOM, ID: 6)
Engine: cisco (CISCO_L7_GLOBAL, ID: 13)

```

Device# **show flow exporter option application table**

```

Engine: prot (IANA_L3_STANDARD, ID: 1)
appID  Name      Description
-----
Engine: port (IANA_L4_STANDARD, ID: 3)
appID  Name      Description
-----

Engine: NBAR (NBAR_CUSTOM, ID: 6)
appID  Name      Description
-----
6:28202 appexample1 User defined protocol appexample1

Engine: cisco (CISCO_L7_GLOBAL, ID: 13)
appID  Name      Description
-----
13:0   unclassified Unclassified traffic
13:1   unknown     Unknown application
13:518 appexample2 appexample2, social web application and service

```

Monitoring AVC with DNS-AS

To display the various AVC with DNS-AS settings you have configured, use these commands in the privileged EXEC mode:

Table 181: AVC with DNS-AS Monitoring Commands

Command	Purpose	Sample Output
show avc dns-as client status	Displays current status of the DNS-AS client. Use this command to know whether AVC with DNS-AS is enabled or not.	Example: show avc dns-as client status
show avc dns-as client trusted-domains	Displays list of trusted domains maintained in the binding table.	Example: show avc dns-as client trusted-domains

Command	Purpose	Sample Output
show avc dns-as client binding-table and show avc dns-as client binding-table detail	Displays AVC with DNS-AS metadata for the list of trusted domains and resolved entries. You can filter the output by application name, domain name, and so on. Both commands display the same information, in different formats.	Example: show avc dns-as client binding-table
show avc dns-as client statistics	Displays packet logging information—the number of DNS queries sent and the number of responses received.	Example: show avc dns-as client statistics
show avc dns-as client name-server brief	Displays information about the DNS server to which the metadata request was sent.	Example: show avc dns-as client name-server brief
show ip name-server	Displays all the name server IP addresses that have been maintained.	Example: show ip name-server
show platform tcam utilization	Displays information about TCAM availability	Example: show platform tcam utilization

Example: show avc dns-as client status

```
Device# show avc dns-as client status
DNS-AS client is enabled
```

Back to [Table 181: AVC with DNS-AS Monitoring Commands](#)

Example: show avc dns-as client trusted-domains

```
Device# show avc dns-as client trusted-domains
Id | Trusted domain
-----
1| example.com
2| www.example.com
3| example.net
4| www.example.net
5| example.org
6| www.example.org
```

Back to [Table 181: AVC with DNS-AS Monitoring Commands](#)

Example: show avc dns-as client binding-table

```
Device# show avc dns-as client binding-table
Switch# show avc dns-as client binding-table detailed
DNS-AS generated protocols:
Max number of protocols :50
Customization interval [min] :N/A

Age : The amount of time that the entry is active
TTL : Time to live which was learned from DNS-AS server
Time To Expire : Entry expiration time in case device does not see DNS traffic for the entry
```



```

host

Protocol-Name : example
VRF : <default>
Host : www.example.com
Age[min] : 2
TTL[min] : 60
Time To Expire[min] : 58
TXT Record : app-name:example|app-class:VO|business:YES
Traffic Class : voip-telephony
Business Relevance : business relevant
IP : 192.0.2.121
   : 192.0.2.254
   : 198.51.100.1
   : 198.51.100.254
   : 192.51.100.12
   : 203.0.113.125
<output truncated>

```

Back to [Table 181: AVC with DNS-AS Monitoring Commands](#)

Example: show avc dns-as client statistics



Note Two DNS servers are configured in this example.

```

Device# show avc dns-as client statistics
Server details: vrf-id = 0 vrf-name = <default> ip = 192.0.2.1
AAAA Query Error packets 0
AAAA Query TX packets 0
AAAA Response RX packets 0
TXT Query Error packets 0
TXT Query TX packets 8
TXT Response RX packets 0
A Query Error packets 0
A Query TX packets 6
A Response RX packets 0
Server details: vrf-id = 0 vrf-name = <default> ip = 192.0.2.2
AAAA Query Error packets 0
AAAA Query TX packets 0
AAAA Response RX packets 0
TXT Query Error packets 0
TXT Query TX packets 2
TXT Response RX packets 2
A Query Error packets 0
A Query TX packets 4
A Response RX packets 2
Total Drop packets 0

avc_dns_as_pkts_logged = 2
avc_dns_as_q_pkts_processed = 2

```

Back to [Table 181: AVC with DNS-AS Monitoring Commands](#)

Example: show avc dns-as client name-server brief

```

Device# show avc dns-as client name-server brief

Server-IP | Vrf-name
-----

```

```
192.0.2.1 | <default>
192.0.2.2 | <default>
```

Back to [Table 181: AVC with DNS-AS Monitoring Commands](#)

Example: show ip name-server

```
Device# show ip name-server
192.0.2.1
192.0.2.2
2001:DB8::1
```

Back to [Table 181: AVC with DNS-AS Monitoring Commands](#)

Example: show platform tcam utilization



Note The relevant TCAM entry is IPv4 qos aces:

```
Device# show platform tcam utilization
CAM Utilization for ASIC# 0 Max Used
Masks/Values Masks/values

Unicast mac addresses: 16604/16604 24/24
IPv4 IGMP groups + multicast routes: 1072/1072 3/3
IPv4 unicast directly-connected routes: 4096/4096 4/4
IPv4 unicast indirectly-connected routes: 1280/1280 40/40
IPv6 Multicast groups: 1072/1072 18/18
IPv6 unicast directly-connected routes: 4096/4096 1/1
IPv6 unicast indirectly-connected routes: 1280/1280 32/32
IPv4 policy based routing aces: 512/512 14/14
IPv4 qos aces: 512/512 51/51
IPv4 security aces: 1024/1024 78/78
IPv6 policy based routing aces: 256/256 8/8
IPv6 qos aces: 256/256 44/44
IPv6 security aces: 512/512 18/18
```

Note: Allocation of TCAM entries per feature uses a complex algorithm. The above information is meant to provide an abstract view of the current TCAM utilization

Back to [Table 181: AVC with DNS-AS Monitoring Commands](#)

Troubleshooting AVC with DNS-AS

Problem	Possible Causes and Solutions
There are no entries in the binding table.	<p>The binding table may be empty because of either one or both of these reasons:</p> <ul style="list-style-type: none"> • Metadata is not maintained in DNS server—complete task Generating Metadata Streams, on page 1880 • The entry is not maintained in the trusted domain list—complete task Maintaining the List of Trusted Domains, on page 1883

Problem	Possible Causes and Solutions
Unsuccessful DNS snooping or packet logging.	To ensure DNS snooping and packet logging, you must attach the policy map (containing the relevant class maps that will determine traffic class) to the interface—See the example in the Configuring QoS for AVC with DNS-AS, on page 1884
The DNS server does not return correct values.	<p>Verify that the correct DNS-AS metadata is maintained in the DNS system.</p> <ul style="list-style-type: none"> Using Linux dig: <pre>dig TXT +short www.example.org [dns-server-ip] "CISCO-CLS=app-name:example app-class:TD business:YES app-id:CU/28202"</pre> Using Windows nslookup: <pre>C:\Windows\system32>NSLookup.exe -q=TXT www.example.org [dns-server-ip] www.example.org text = "CISCO-CLS=app-name:example app-class:TD business:YES app-id:CU/28202"</pre>
The QoS policy you applied is removed from the port.	<p>When the DNS-AS client recognises an application, along with saving the "A" record response in the binding table, the system utilises the TCAM to save the IP address of the application. A single application can in effect have multiple IP addresses, each utilising additional space in the TCAM. When the TCAM is exhausted, QoS policies cease to be applied.</p> <p>To avoid the problem, monitor TCAM utilisation on a regular basis. Enter the show platform tcam utilisation command in privilege EXEC mode, to display information about TCAM availability.</p>
The DNS-AS client ignores the QoS mapping you've defined and applies default forwarding behavior.	<p>The DNS-AS client ignores QoS mapping and applies default forwarding behavior in these cases:</p> <ul style="list-style-type: none"> If the match attributes that you specify for the traffic class and business relevance do not match what you have defined in the metadata stream—Check and correct as required. If the binding table entry is no longer active. This refers to the age of the entry—Use the show avc dns-as client binding-table command to display the age of an entry.

Feature History and Information for AVC with DNS-AS

The following table provides release information about the feature or features described in this chapter. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Release	Modification
Cisco IOS Release 15.2(5)E1	This feature was introduced.

Release	Modification
Cisco IOS Release 15.2(5)E2	Flexible NetFlow (FNF) for AVC with DNS-AS was introduced - Provides the ability to export application information using FNF.



CHAPTER 94

Configuring SDM Templates

- [Information About Configuring SDM Templates, on page 1899](#)
- [How to Configure SDM Templates, on page 1901](#)
- [Configuration Examples for SDM Templates, on page 1903](#)
- [Additional References for SDM Templates, on page 1904](#)
- [Feature History and Information for Configuring SDM Templates, on page 1905](#)

Information About Configuring SDM Templates

Restrictions for SDM Templates

The following are restrictions when using SDM templates:

The switch supports homogeneous stacking, but does not support mixed stacking.

SDM Templates

You can use Switch Database Management (SDM) templates to configure system resources to optimize support for specific features, depending on how your device is used in the network. You can select a template to provide maximum system usage for some functions: for example, use the default template to balance resources, and use the access template to obtain maximum ACL usage. The switch SDM templates allocate system hardware resources for different uses.

You can select SDM templates for IP Version 4 (IPv4) to optimize these features on switches running the IP Lite feature set:



Note When the switch is running the LAN Base feature set, you must use the default template.

- IPv4—The IPv4 template provides maximum support for IPv4. No IPv6 support is available on this template.
- VLAN—The VLAN template provides Layer 2 functionality and the maximum number of unicast MAC addresses.

- Default—The default template gives balance to all functions providing support for Layer 2, IPv4 and IPv6 functionality.



Note Use the default template when configuring IPv4 static routing on SVIs on switches running the LAN Base feature set. You can configure up to 16 static routes.

For information about homogeneous stacks, see the *Catalyst 2960-XR Switch Stacking Configuration Guide*.

After you change the template and the system reboots, you can use the **show sdm prefer** privileged EXEC command to verify the new template configuration. If you enter the **show sdm prefer** command before you enter the **reload** privileged EXEC command, the **show sdm prefer** command shows the template currently in use and the template that becomes active after a reload.



Note

- The SDM templates contain only those commands that are defined as part of the templates. If a template enables another related command that is not defined in the template, then this other command will be visible when the **show running config** command is entered. For example, if the SDM template enables the **switchport voice vlan** command, then the **spanning-tree portfast edge** command may also be enabled (although it is not defined on the SDM template).

If the SDM template is removed, then other such related commands are also removed and have to be reconfigured explicitly.

- SDM templates do not create VLANs. You must create the VLANs before adding commands to the SDM templates.

SDM Template Resources

You can use SDM templates to configure system resources to optimize support for specific features, depending on how your device is used in the network. You can select a template to provide maximum system usage for some functions.

To allocate ternary content addressable memory (TCAM) resources for different usages, the switch SDM templates prioritize system resources to optimize support for certain features. These templates are supported on your device:

Table 182: Approximate Number of Feature Resources Allowed by Templates

Resource	Default	IPv4	VLAN
Unicast MAC addresses	16 K	16 K	32 K
Active VLANs/VLAN IDs	1K/4096	1K/4096	1K/4096
IPv4 IGMP groups	1 K	1 K	1 K
IPv4 unicast routes	5.25 K	24 K	0.5 K

Resource	Default	IPv4	VLAN
• Directly connected hosts	4 K	16 K	0.25 K
• Indirect routes	1.25 K	8 K	0.25 K
IPv4 policy-based routing ACEs	0.5 K	0.375 k	0.5 K
IPv6 multicast groups:	1 K	0	1 K
• Directly connected IPv6 addresses	4 K	0	0.25 K
• Indirect IPv6 unicast routes	1.25 K	0	0.25 K
IPv4 MAC QoS ACEs	0.5 K	0.5 K	0.5 K
IPv4 MAC security ACEs	1 K	0.875 k	1 K
IPv6 policy-based routing ACEs	0.25 K	0	0
IPv6 QoS ACEs	0.25 K	0	0.5 K
IPv6 security ACEs	0.5 K	60	0.5 K

SDM Templates and Switch Stacks

All stack members use the same SDM template that is stored on the active stack. When a new switch is added to a stack, as with the switch configuration and VLAN database files, the SDM configuration that is stored on the active stack overrides the template configured on an individual switch.

Version-mismatch (VM) mode has priority over SDM-mismatch mode. If a VM mode condition and an SDM-mismatch mode exist, the switch stack first attempts to resolve the VM-mode condition. You can use the **show switch** privileged EXEC command to see if any stack members are in SDM-mismatch mode.

How to Configure SDM Templates

Setting the SDM Template

Follow these steps to use the SDM template to maximize feature usage:

SUMMARY STEPS

1. enable
2. configure terminal
3. sdm prefer { default | ipv4 | vlan }
4. end
5. reload

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p>enable</p> <p>Example:</p> <pre>Device> enable</pre>	<p>Enables privileged EXEC mode.</p> <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	<p>configure terminal</p> <p>Example:</p> <pre>Device# configure terminal</pre>	<p>Enters global configuration mode.</p>
Step 3	<p>sdm prefer { default ipv4 vlan }</p> <p>Example:</p> <pre>Device(config)# sdm prefer default</pre>	<p>Specifies the SDM template to be used on the switch. The keywords have these meanings:</p> <ul style="list-style-type: none"> • default —The default template provides balance for all Layer 2, IPv4 and IPv6 functionality. • IPv4 —The IPv4 template provides maximum support for IPv4. No IPv6 support is available on this template. • VLAN —The VLAN template provides Layer 2 functionality and the maximum number of unicast MAC addresses. <p>Use the no sdm prefer command to set the switch to the default template. The default template balances the use of system resources.</p>
Step 4	<p>end</p> <p>Example:</p> <pre>Device(config)# end</pre>	<p>Returns to privileged EXEC mode.</p>
Step 5	<p>reload</p> <p>Example:</p> <pre>Device# reload</pre>	<p>Reloads the operating system.</p>

Configuration Examples for SDM Templates

Examples: Displaying SDM Templates

This is an example output showing the default template information:

```
Device# show sdm prefer default

"default" template:
The selected template optimizes the resources in
the switch to support this level of features for
8 routed interfaces and 1024 VLANs.

number of unicast mac addresses:          16K
number of IPv4 IGMP groups + multicast routes: 1K
number of IPv4 unicast routes:           5.25K
  number of directly-connected IPv4 hosts: 4K
  number of indirect IPv4 routes:         1.25K
number of IPv6 multicast groups:          1K
number of directly-connected IPv6 addresses: 4K
number of indirect IPv6 unicast routes:    1.25K
number of IPv4 policy based routing aces: 0.5K
number of IPv4/MAC qos aces:              0.5K
number of IPv4/MAC security aces:         1K
number of IPv6 policy based routing aces: 0.25K
number of IPv6 qos aces:                  0.25K
number of IPv6 security aces:             0.5K

Device#
```

This is an example output showing the IPv4 template information:

```
Device# show sdm prefer ipv4

"ipv4" template:
The selected template optimizes the resources in
the switch to support this level of features for
8 routed interfaces and 1024 VLANs.

number of unicast mac addresses:          16K
number of IPv4 IGMP groups + multicast routes: 1K
number of IPv4 unicast routes:           24K
  number of directly-connected IPv4 hosts: 16K
  number of indirect IPv4 routes:         8K
number of IPv6 multicast groups:          0
number of directly-connected IPv6 addresses: 0
number of indirect IPv6 unicast routes:    0
number of IPv4 policy based routing aces: 0.375k
number of IPv4/MAC qos aces:              0.5K
number of IPv4/MAC security aces:         0.875k
number of IPv6 policy based routing aces: 0
number of IPv6 qos aces:                  0
number of IPv6 security aces:             60

Device#
```

This is an example output showing the VLAN template information:

```

Device# show sdm prefer vlan

"vlan" template:
The selected template optimizes the resources in
the switch to support this level of features for
8 routed interfaces and 1024 VLANs.

number of unicast mac addresses:          32K
number of IPv4 IGMP groups + multicast routes: 1K
number of IPv4 unicast routes:           0.5K
  number of directly-connected IPv4 hosts: 0.25K
  number of indirect IPv4 routes:         0.25K
number of IPv6 multicast groups:          1K
number of directly-connected IPv6 addresses: 0.25K
number of indirect IPv6 unicast routes:    0.25K
number of IPv4 policy based routing aces: 0.5K
number of IPv4/MAC qos aces:              0.5K
number of IPv4/MAC security aces:         1K
number of IPv6 policy based routing aces:  0
number of IPv6 qos aces:                  0.5K
number of IPv6 security aces:             0.5K

```

Examples: Configuring SDM Templates

This example shows how to configure the VLAN template:

```

Device(config)# sdm prefer vlan
Device(config)# exit
Device# reload
  Proceed with reload? [confirm]

```

Additional References for SDM Templates

Related Documents

Related Topic	Document Title
SDM commands	<i>Catalyst 2960-XR Switch System Management Command Reference</i>

Standards and RFCs

Standard/RFC	Title
None	—

MIBs

MIB	MIBs Link
All the supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
<p>The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies.</p> <p>To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds.</p> <p>Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.</p>	http://www.cisco.com/support

Feature History and Information for Configuring SDM Templates

Release	Modification
Cisco IOS Release 15.0(2)EX1	This feature was introduced.



CHAPTER 95

Configuring System Message Logging and Smart Logging

- [Information About Configuring System Message Logs and Smart Logs, on page 1907](#)
- [How to Configure System Message Logs and Smart Logs, on page 1910](#)
- [Monitoring and Maintaining System Message Logs and Smart Logs, on page 1922](#)
- [Configuration Examples for System Message Logs and Smart Logs, on page 1923](#)
- [Additional References for System Message Logs and Smart Logs, on page 1924](#)
- [Feature History and Information For System Message Logs, on page 1925](#)

Information About Configuring System Message Logs and Smart Logs

System Message Logging

By default, a switch sends the output from system messages and **debug** privileged EXEC commands to a logging process. . The logging process controls the distribution of logging messages to various destinations, such as the logging buffer, terminal lines, or a UNIX syslog server, depending on your configuration. The process also sends messages to the console.

When the logging process is disabled, messages are sent only to the console. The messages are sent as they are generated, so message and debug output are interspersed with prompts or output from other commands. Messages appear on the active consoles after the process that generated them has finished.

You can set the severity level of the messages to control the type of messages displayed on the consoles and each of the destinations. You can time-stamp log messages or set the syslog source address to enhance real-time debugging and management. For information on possible messages, see the system message guide for this release.

You can access logged system messages by using the switch command-line interface (CLI) or by saving them to a properly configured syslog server. The switch software saves syslog messages in an internal buffer on a standalone switch. If a standalone switch , the log is lost unless you had saved it to flash memory.

You can remotely monitor system messages by viewing the logs on a syslog server or by accessing the switch through Telnet, through the console port, or through the Ethernet management port.



Note The syslog format is compatible with 4.3 BSD UNIX.

System Log Message Format

System log messages can contain up to 80 characters and a percent sign (%), which follows the optional sequence number or time-stamp information, if configured. Depending on the switch, messages appear in one of these formats:

- *seq no:timestamp: %facility-severity-MNEMONIC:description (hostname-n)*
- *seq no:timestamp: %facility-severity-MNEMONIC:description*

The part of the message preceding the percent sign depends on the setting of these global configuration commands:

- **service sequence-numbers**
- **service timestamps log datetime**
- **service timestamps log datetime [localtime] [msec] [show-timezone]**
- **service timestamps log uptime**

Table 183: System Log Message Elements

Element	Description
<i>seq no:</i>	Stamps log messages with a sequence number only if the service sequence-numbers global configuration command is configured.
<i>timestamp</i> formats: <i>mm/dd h h:mm:ss</i> or <i>hh:mm:ss</i> (short uptime) or <i>d h</i> (long uptime)	Date and time of the message or event. This information appears only if the service timestamps log [datetime log] global configuration command is configured.
<i>facility</i>	The facility to which the message refers (for example, SNMP, SYS, and so forth).
<i>severity</i>	Single-digit code from 0 to 7 that is the severity of the message.
<i>MNEMONIC</i>	Text string that uniquely describes the message.
<i>description</i>	Text string containing detailed information about the event being reported.

Default System Message Logging Settings

Table 184: Default System Message Logging Settings

Feature	Default Setting
System message logging to the console	Enabled.
Console severity	Debugging.
Logging file configuration	No filename specified.
Logging buffer size	4096 bytes.
Logging history size	1 message.
Time stamps	Disabled.
Synchronous logging	Disabled.
Logging server	Disabled.
Syslog server IP address	None configured.
Server facility	Local7
Server severity	Informational.

Syslog Message Limits

If you enabled syslog message traps to be sent to an SNMP network management station by using the **snmp-server enable trap** global configuration command, you can change the level of messages sent and stored in the switch history table. You also can change the number of messages that are stored in the history table.

Messages are stored in the history table because SNMP traps are not guaranteed to reach their destination. By default, one message of the level **warning** and numerically lower levels are stored in the history table even if syslog traps are not enabled.

When the history table is full (it contains the maximum number of message entries specified with the **logging history size** global configuration command), the oldest message entry is deleted from the table to allow the new message entry to be stored.

The history table lists the level keywords and severity level. For SNMP usage, the severity level values increase by 1. For example, *emergencies* equal 1, not 0, and *critical* equals 3, not 2.

Smart Logging

Smart logging provides a mechanism to capture and export packet flows based on predefined or user-configured triggers. The switch supports smart logging for these events:

- DHCP snooping violations

- Dynamic ARP inspection violations
- IP source guard denied traffic
- ACL permitted or denied traffic

To use smart logging, you must first configure a NetFlow exporter that you identify when you enable smart logging. For information on configuring the NetFlow feature, see the *Catalyst 2960-XR Switch NetFlow Lite Configuration Guide*.

Smart logging processing creates a NetFlow packet for the configured event and sends the packet to the external NetFlow collector. Smart logging counters reflect the number of packets that are logged. This number is the same as the number of packets sent to the collector if no packets are dropped between the switch and the NetFlow collector. You enable smart logging globally on the switch, and you can then configure specific events to be smart logged.

Smart Logging for Port ACL Deny or Permit Actions

The switch supports port ACLs, router ACLs, and VLAN ACLs.

- Port ACLs are IP or MAC ACLs applied to a Layer 2 port. Logging is not supported on port ACLs, but smart logging is supported on IP ACLs applied to Layer 2 ports.
- Router ACLs are ACLs applied to Layer 3 ports. Router ACLs support logging but not smart logging.
- VLAN ACLs or VLAN maps are ACLs applied to VLANs. You can configure logging on VLAN maps, but not smart logging.

When you configure any permit or deny ACL, you can configure logging or smart logging as part of the access list, to take place on all traffic that the ACL permits or denies. The type of port that you attach the ACL to determines the type of logging. If you attach an ACL with smart log configured to a router or a VLAN, the ACL is attached, but smart logging does not take affect. If you configure logging on an ACL attached to a Layer 2 port, the logging keyword is ignored.

How to Configure System Message Logs and Smart Logs

Setting the Message Display Destination Device

If message logging is enabled, you can send messages to specific locations in addition to the console.

This task is optional.

SUMMARY STEPS

1. **configure terminal**
2. **logging buffered** *[size]*
3. **logging host**
4. **logging file flash:** *filename* *[max-file-size [min-file-size]]* *[severity-level-number | type]*
5. **end**
6. **terminal monitor**

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p>configure terminal</p> <p>Example:</p> <pre>Device# configure terminal</pre>	Enters global configuration mode.
Step 2	<p>logging buffered <i>[size]</i></p> <p>Example:</p> <pre>Device(config)# logging buffered 8192</pre>	<p>Logs messages to an internal buffer on the switch or on a standalone switch or, in the case of a switch stack, on the active switch. The range is 4096 to 2147483647 bytes. The default buffer size is 4096 bytes.</p> <p>If a standalone switch or the active switch fails, the log file is lost unless you previously saved it to flash memory. See Step 4.</p> <p>Note Do not make the buffer size too large because the switch could run out of memory for other tasks. Use the show memory privileged EXEC command to view the free processor memory on the switch. However, this value is the maximum available, and the buffer size should <i>not</i> be set to this amount.</p>
Step 3	<p>logging <i>host</i></p> <p>Example:</p> <pre>Device(config)# logging 125.1.1.100</pre>	<p>Logs messages to a UNIX syslog server host.</p> <p><i>host</i> specifies the name or IP address of the host to be used as the syslog server.</p> <p>To build a list of syslog servers that receive logging messages, enter this command more than once.</p>
Step 4	<p>logging file flash: <i>filename</i> [<i>max-file-size</i> [<i>min-file-size</i>]] [<i>severity-level-number</i> <i>type</i>]</p> <p>Example:</p> <pre>Device(config)# logging file flash:log_msg.txt 40960 4096 3</pre>	<p>Stores log messages in a file in flash memory on a standalone switch or, in the case of a switch stack, on the active switch.</p> <ul style="list-style-type: none"> • <i>filename</i>—Enters the log message filename. • (Optional) max-file-size —Specifies the maximum logging file size. The range is 4096 to 2147483647. The default is 4096 bytes. • (Optional) <i>min-file-size</i>—Specifies the minimum logging file size. The range is 1024 to 2147483647. The default is 2048 bytes. • (Optional) <i>severity-level-number</i> <i>type</i>—Specifies either the logging severity level or the logging type. The severity range is 0 to 7.
Step 5	<p>end</p> <p>Example:</p>	Returns to privileged EXEC mode.

	Command or Action	Purpose
	Device(config)# end	
Step 6	terminal monitor Example: Device# terminal monitor	Logs messages to a nonconsole terminal during the current session. Terminal parameter-setting commands are set locally and do not remain in effect after the session has ended. You must perform this step for each session to see the debugging messages.

Synchronizing Log Messages

You can synchronize unsolicited messages and **debug** privileged EXEC command output with solicited device output and prompts for a specific console port line or virtual terminal line. You can identify the types of messages to be output asynchronously based on the level of severity. You can also configure the maximum number of buffers for storing asynchronous messages for the terminal after which messages are dropped.

When synchronous logging of unsolicited messages and **debug** command output is enabled, unsolicited device output appears on the console or printed after solicited device output appears or is printed. Unsolicited messages and **debug** command output appears on the console after the prompt for user input is returned. Therefore, unsolicited messages and **debug** command output are not interspersed with solicited device output and prompts. After the unsolicited messages appear, the console again displays the user prompt.

This task is optional.

SUMMARY STEPS

1. **configure terminal**
2. **line** [**console** | **vty**] *line-number* [*ending-line-number*]
3. **logging synchronous** [**level** [*severity-level* | **all**] | **limit** *number-of-buffers*]
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 2	line [console vty] <i>line-number</i> [<i>ending-line-number</i>] Example: Device(config)# line console	Specifies the line to be configured for synchronous logging of messages. <ul style="list-style-type: none"> • console —Specifies configurations that occur through the switch console port or the Ethernet management port.

	Command or Action	Purpose
		<ul style="list-style-type: none"> • line vty <i>line-number</i>—Specifies which vty lines are to have synchronous logging enabled. You use a vty connection for configurations that occur through a Telnet session. The range of line numbers is from 0 to 15. <p>You can change the setting of all 16 vty lines at once by entering:</p> <pre>line vty 0 15</pre> <p>You can also change the setting of the single vty line being used for your current connection. For example, to change the setting for vty line 2, enter:</p> <pre>line vty 2</pre> <p>When you enter this command, the mode changes to line configuration.</p>
Step 3	<p>logging synchronous [level [<i>severity-level</i> all] limit <i>number-of-buffers</i>]</p> <p>Example:</p> <pre>Device(config)# logging synchronous level 3 limit 1000</pre>	<p>Enables synchronous logging of messages.</p> <ul style="list-style-type: none"> • (Optional) level <i>severity-level</i>—Specifies the message severity level. Messages with a severity level equal to or higher than this value are printed asynchronously. Low numbers mean greater severity and high numbers mean lesser severity. The default is 2. • (Optional) level all—Specifies that all messages are printed asynchronously regardless of the severity level. • (Optional) limit <i>number-of-buffers</i>—Specifies the number of buffers to be queued for the terminal after which new messages are dropped. The range is 0 to 2147483647. The default is 20.
Step 4	<p>end</p> <p>Example:</p> <pre>Device(config)# end</pre>	<p>Returns to privileged EXEC mode.</p>

Disabling Message Logging

Message logging is enabled by default. It must be enabled to send messages to any destination other than the console. When enabled, log messages are sent to a logging process, which logs messages to designated locations asynchronously to the processes that generated the messages.

Disabling the logging process can slow down the switch because a process must wait until the messages are written to the console before continuing. When the logging process is disabled, messages appear on the console as soon as they are produced, often appearing in the middle of command output.

The **logging synchronous** global configuration command also affects the display of messages to the console. When this command is enabled, messages appear only after you press **Return**.

To reenable message logging after it has been disabled, use the **logging on** global configuration command.

This task is optional.

SUMMARY STEPS

1. **configure terminal**
2. **no logging console**
3. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 2	no logging console Example: Device(config)# no logging console	Disables message logging.
Step 3	end Example: Device(config)# end	Returns to privileged EXEC mode.

Enabling and Disabling Time Stamps on Log Messages

By default, log messages are not time-stamped.

This task is optional.

SUMMARY STEPS

1. **configure terminal**
2. Use one of these commands:
 - **service timestamps log uptime**
 - **service timestamps log datetime[msec | localtime | show-timezone]**
3. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode.
Step 2	Use one of these commands: <ul style="list-style-type: none"> • service timestamps log uptime • service timestamps log datetime[msec localtime show-timezone] Example: Device(config)# <code>service timestamps log uptime</code> or Device(config)# <code>service timestamps log datetime</code>	Enables log time stamps. <ul style="list-style-type: none"> • log uptime—Enables time stamps on log messages, showing the time since the system was rebooted. • log datetime—Enables time stamps on log messages. Depending on the options selected, the time stamp can include the date, time in milliseconds relative to the local time zone, and the time zone name.
Step 3	end Example: Device(config)# <code>end</code>	Returns to privileged EXEC mode.

Enabling and Disabling Sequence Numbers in Log Messages

If there is more than one log message with the same time stamp, you can display messages with sequence numbers to view these messages. By default, sequence numbers in log messages are not displayed.

This task is optional.

SUMMARY STEPS

1. `configure terminal`
2. `service sequence-numbers`
3. `end`

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode.

	Command or Action	Purpose
Step 2	service sequence-numbers Example: Device(config) # service sequence-numbers	Enables sequence numbers.
Step 3	end Example: Device(config) # end	Returns to privileged EXEC mode.

Defining the Message Severity Level

Limit messages displayed to the selected device by specifying the severity level of the message.

This task is optional.

SUMMARY STEPS

1. **configure terminal**
2. **logging console** *level*
3. **logging monitor** *level*
4. **logging trap** *level*
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 2	logging console <i>level</i> Example: Device(config) # logging console 3	Limits messages logged to the console. By default, the console receives debugging messages and numerically lower levels.
Step 3	logging monitor <i>level</i> Example: Device(config) # logging monitor 3	Limits messages logged to the terminal lines. By default, the terminal receives debugging messages and numerically lower levels.

	Command or Action	Purpose
Step 4	logging trap <i>level</i> Example: Device(config)# logging trap 3	Limits messages logged to the syslog servers. By default, syslog servers receive informational messages and numerically lower levels.
Step 5	end Example: Device(config)# end	Returns to privileged EXEC mode.

Limiting Syslog Messages Sent to the History Table and to SNMP

This task explains how to limit syslog messages that are sent to the history table and to SNMP.

This task is optional.

SUMMARY STEPS

1. **configure terminal**
2. **logging history *level***
3. **logging history size *number***
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 2	logging history <i>level</i> Example: Device(config)# logging history 3	Changes the default level of syslog messages stored in the history file and sent to the SNMP server. By default, warnings , errors , critical , alerts , and emergencies messages are sent.
Step 3	logging history size <i>number</i> Example: Device(config)# logging history size 200	Specifies the number of syslog messages that can be stored in the history table. The default is to store one message. The range is 0 to 500 messages.

	Command or Action	Purpose
Step 4	end Example: Device(config)# end	Returns to privileged EXEC mode.

Logging Messages to a UNIX Syslog Daemon

This task is optional.



Note Some recent versions of UNIX syslog daemons no longer accept by default syslog packets from the network. If this is the case with your system, use the UNIX **man syslogd** command to decide what options must be added to or removed from the syslog command line to enable logging of remote syslog messages.

Before you begin

- Log in as root.
- Before you can send system log messages to a UNIX syslog server, you must configure the syslog daemon on a UNIX server.

SUMMARY STEPS

1. Add a line to the file `/etc/syslog.conf`.
2. Enter these commands at the UNIX shell prompt.
3. Make sure the syslog daemon reads the new changes.

DETAILED STEPS

	Command or Action	Purpose
Step 1	Add a line to the file <code>/etc/syslog.conf</code> . Example: <pre>local7.debug /usr/adm/logs/cisco.log</pre>	<ul style="list-style-type: none"> • local7—Specifies the logging facility. • debug—Specifies the syslog level. The file must already exist, and the syslog daemon must have permission to write to it.
Step 2	Enter these commands at the UNIX shell prompt. Example: <pre>\$ touch /var/log/cisco.log \$ chmod 666 /var/log/cisco.log</pre>	Creates the log file. The syslog daemon sends messages at this level or at a more severe level to this file.
Step 3	Make sure the syslog daemon reads the new changes. Example:	For more information, see the man syslog.conf and man syslogd commands on your UNIX system.

	Command or Action	Purpose
	<code>\$ kill -HUP `cat /etc/syslog.pid`</code>	

Enabling Smart Logging

SUMMARY STEPS

1. configure terminal
2. logging smartlog
3. logging smartlog exporter *exporter_name*
4. logging packet capture size *packet_size*
5. end

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode.
Step 2	logging smartlog Example: Device(config)# <code>logging smartlog</code>	Turns on the smart logging feature.
Step 3	logging smartlog exporter <i>exporter_name</i> Example: Device(config)# <code>logging smartlog exporter export-file</code>	Identifies the smart log exporter. You must have already configured the exporter by using the NetFlow CLI. If the exporter name does not exist, you receive an error message. By default, the switch sends data to the collector every 60 seconds.
Step 4	logging packet capture size <i>packet_size</i> Example: Device(config)# <code>logging packet capture size 128</code>	(Optional) Configures the size of the packet to be sent to the exporter. The range is from 64 to 1024 bytes in 4-byte increments. The default size is 64 bytes. Note Increasing the packet capture size reduces the number of flow records per packet.
Step 5	end Example: Device(config)# <code>end</code>	Returns to privileged EXEC mode.

Enabling Smart Logging for DHCP Snooping Violations

DHCP snooping intercepts and inspects DHCP packets entering untrusted ports and either forwards or drops the packets. You can enable DHCP snooping smart logging to send the contents of dropped packets to the NetFlow collector.

SUMMARY STEPS

1. **configure terminal**
2. **ip dhcp snooping vlan** {*vlan-id* | *vlan-range*} **smartlog**
3. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 2	ip dhcp snooping vlan { <i>vlan-id</i> <i>vlan-range</i> } smartlog Example: Device(config)# ip dhcp snooping vlan 5-8 smartlog	Specifies a VLAN ID or a range of VLANs on which to enable DHCP snooping smart logging.
Step 3	end Example: Device(config)# end	Returns to privileged EXEC mode.

Enabling Smart Logging for Dynamic ARP Inspection Violations

Dynamic ARP inspection intercepts ARP packets on untrusted ports and validates them before forwarding. The functionality is similar to DHCP snooping but for ARP packets. You can configure dynamic ARP inspection logging by using the **ip arp inspection log-buffer** global configuration command. By default, all dropped packets are logged. You can also configure the switch to apply smart logging to the same packets that are being logged, sending the packet contents packet to the Cisco NetFlow collector.

SUMMARY STEPS

1. **configure terminal**
2. **ip arp inspection smartlog**
3. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode.
Step 2	ip arp inspection smartlog Example: Device(config)# <code>ip arp inspection smartlog</code>	Specifies that whatever packets are currently being logged (the default is all dropped packets) are also smart-logged.
Step 3	end Example: Device(config)# <code>end</code>	Returns to privileged EXEC mode.

Enabling Smart Logging for IP Source Guard Violations

IP source guard is a security feature related to DHCP snooping. You can use IP source guard to filter traffic based on the IP source address or the MAC address. All IP packets with a source address other than the specified address or addresses learned through DHCP snooping are denied. You can enable IP source guard smart logging to send the contents of the denied packets to the NetFlow collector.

SUMMARY STEPS

1. `configure terminal`
2. `interface interface-id`
3. `ip verify source smartlog`
4. `end`

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example: Device# <code>configure terminal</code>	Enters global configuration mode.
Step 2	interface interface-id Example: Device(config)# <code>interface GigabitEthernet1/0/1</code>	Specifies an interface and enters interface configuration mode.

	Command or Action	Purpose
Step 3	ip verify source smartlog Example: <pre>Device(config-if)# ip verify source smartlog</pre>	Enables IP source guard smart logging for all packets that are denied by IP source guard.
Step 4	end Example: <pre>Device(config-if)# end</pre>	Returns to privileged EXEC mode.

Monitoring and Maintaining System Message Logs and Smart Logs

Monitoring Configuration Archive Logs

Command	Purpose
show archive log config { all number [<i>end-number</i>] user <i>username</i> [session number] <i>number</i> [<i>end-number</i>] statistics } [provisioning]	Displays the entire configuration log or the log for specified parameters.

Monitoring Smart Logging

Command	Purpose
show logging smartlog	Displays smart logging entries.
show ip arp inspection	Displays the IP ARP smart logging entries.
show ip verify source	Displays IP source guard smart logging entries. The output shows whether or not smart logging is enabled on the interface.

Configuration Examples for System Message Logs and Smart Logs

Example: Stacking System Message

This example shows a partial switch system message for active stack and a stack member (hostname *Switch-2*):

```
00:00:46: %LINK-3-UPDOWN: Interface Port-channell, changed state to up
00:00:47: %LINK-3-UPDOWN: Interface GigabitEthernet1/0/1, changed state to up
00:00:47: %LINK-3-UPDOWN: Interface GigabitEthernet1/0/2, changed state to up
00:00:48: %LINEPROTO-5-UPDOWN: Line protocol on Interface Vlan1, changed state to down
00:00:48: %LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet1/0/1, changed
state to down 2
*Mar  1 18:46:11: %SYS-5-CONFIG_I: Configured from console by vty2 (10.34.195.36)
18:47:02: %SYS-5-CONFIG_I: Configured from console by vty2 (10.34.195.36)
*Mar  1 18:48:50.483 UTC: %SYS-5-CONFIG_I: Configured from console by vty2 (10.34.195.36)

00:00:46: %LINK-3-UPDOWN: Interface Port-channell, changed state to up (Switch-2)
00:00:47: %LINK-3-UPDOWN: Interface GigabitEthernet2/0/1, changed state to up (Switch-2)
00:00:47: %LINK-3-UPDOWN: Interface GigabitEthernet2/0/2, changed state to up (Switch-2)
00:00:48: %LINEPROTO-5-UPDOWN: Line protocol on Interface Vlan1, changed state to down
(Switch-2)
00:00:48: %LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet2/0/1, changed
state to down 2 (Switch-2)
```

Example: Switch System Message

This example shows a partial switch system message on a switch:

```
00:00:46: %LINK-3-UPDOWN: Interface Port-channell, changed state to up
00:00:47: %LINK-3-UPDOWN: Interface GigabitEthernet0/1, changed state to up
00:00:47: %LINK-3-UPDOWN: Interface GigabitEthernet0/2, changed state to up
00:00:48: %LINEPROTO-5-UPDOWN: Line protocol on Interface Vlan1, changed state to down
00:00:48: %LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0/1, changed state
to down 2
*Mar  1 18:46:11: %SYS-5-CONFIG_I: Configured from console by vty2 (10.34.195.36)
18:47:02: %SYS-5-CONFIG_I: Configured from console by vty2 (10.34.195.36)
*Mar  1 18:48:50.483 UTC: %SYS-5-CONFIG_I: Configured from console by vty2 (10.34.195.36)
```

Example: Enabling Smart Logging

You add the smart log configuration option when you create the permit and deny conditions for an ACL.

This example shows how to enable smart logging on a numbered access list:

```
Device(config)# access-list 199 permit ip any any smartlog
```

This example shows how to enable smart logging on a named access list:

```
Device(config)# ip access-list extended test1
Device(config-ext-nacl)# deny ip host 10.1.1.3 any smartlog
```

Examples: Displaying Service Timestamps Log

This example shows part of a logging display with the **service timestamps log datetime** global configuration command enabled:

```
*Mar 1 18:46:11: %SYS-5-CONFIG_I: Configured from console by vty2 (10.34.195.36)
(Switch-2)
```

This example shows part of a logging display with the **service timestamps log uptime** global configuration command enabled:

```
00:00:46: %LINK-3-UPDOWN: Interface Port-channel1, changed state to up (Switch-2)
```

This example shows part of a logging display with the sequence numbers enabled.

```
000019: %SYS-5-CONFIG_I: Configured from console by vty2 (10.34.195.36) (Switch-2)
```

Additional References for System Message Logs and Smart Logs

Related Documents

Related Topic	Document Title
System message log commands	<i>Catalyst 2960-XR Switch System Management Command Reference</i>
Platform-independent command references	<i>Cisco IOS 15.3M&T Command References</i>
Platform-independent configuration information	<i>Cisco IOS 15.3M&T Configuration Guides</i>

Standards and RFCs

Standard/RFC	Title
None	—

MIBs

MIB	MIBs Link
All supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
<p>The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies.</p> <p>To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds.</p> <p>Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.</p>	http://www.cisco.com/support

Feature History and Information For System Message Logs

Release	Modification
Cisco IOS Release 15.0(2)EX1	This feature was introduced.



CHAPTER 96

Configuring Online Diagnostics

- [Information About Configuring Online Diagnostics, on page 1927](#)
- [How to Configure Online Diagnostics, on page 1928](#)
- [Monitoring and Maintaining Online Diagnostics, on page 1932](#)
- [Configuration Examples for Online Diagnostic Tests, on page 1933](#)
- [Additional References for Online Diagnostics, on page 1936](#)
- [Feature History and Information for Configuring Online Diagnostics, on page 1937](#)

Information About Configuring Online Diagnostics

Online Diagnostics

With online diagnostics, you can test and verify the hardware functionality of the Device while the Device is connected to a live network.

The online diagnostics contain packet switching tests that check different hardware components and verify the data path and the control signals.

The online diagnostics detect problems in these areas:

- Hardware components
- Interfaces (Ethernet ports and so forth)
- Solder joints

Online diagnostics are categorized as on-demand, scheduled, or health-monitoring diagnostics. On-demand diagnostics run from the CLI; scheduled diagnostics run at user-designated intervals or at specified times when the Device is connected to a live network; and health-monitoring runs in the background with user-defined intervals. By default, the health-monitoring test runs for every 30 seconds.

After you configure online diagnostics, you can manually start diagnostic tests or display the test results. You can also see which tests are configured for the Device or switch stack and the diagnostic tests that have already run.

How to Configure Online Diagnostics

Starting Online Diagnostic Tests

After you configure diagnostic tests to run on the switch, use the **diagnostic start** privileged EXEC command to begin diagnostic testing.

After starting the tests, you cannot stop the testing process.

Use this privileged EXEC command to manually start online diagnostic testing.

SUMMARY STEPS

1. **diagnostic start switch** *number* **test** {*name* | *test-id* | *test-id-range* | **all** | **basic** | **non-disruptive** }

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p>diagnostic start switch <i>number</i> test {<i>name</i> <i>test-id</i> <i>test-id-range</i> all basic non-disruptive }</p> <p>Example:</p> <pre>Device# diagnostic start switch 2 test basic</pre>	<p>Starts the diagnostic tests.</p> <p>The switch <i>number</i> keyword is supported only on stacking switches. The range is from 1 to 8.</p> <p>You can specify the tests by using one of these options:</p> <ul style="list-style-type: none"> • <i>name</i>—Enters the name of the test. • <i>test-id</i>—Enters the ID number of the test. • <i>test-id-range</i>—Enters the range of test IDs by using integers separated by a comma and a hyphen. • all—Starts all of the tests. • basic— Starts the basic test suite. • non-disruptive—Starts the non-disruptive test suite.

Configuring Online Diagnostics

You must configure the failure threshold and the interval between tests before enabling diagnostic monitoring.

Scheduling Online Diagnostics

You can schedule online diagnostics to run at a designated time of day or on a daily, weekly, or monthly basis for a switch. Use the **no** form of this command to remove the scheduling.

SUMMARY STEPS

1. **configure terminal**

2. **diagnostic schedule switch** *number test* {*name* | *test-id* | *test-id-range* | **all** | **basic** | **non-disruptive** } {**daily** | **on** *mm dd yyyy hh:mm* | **weekly** *day-of-week hh:mm*}

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p>configure terminal</p> <p>Example:</p> <pre>Device# configure terminal</pre>	Enters global configuration mode.
Step 2	<p>diagnostic schedule switch <i>number test</i> {<i>name</i> <i>test-id</i> <i>test-id-range</i> all basic non-disruptive } {daily on <i>mm dd yyyy hh:mm</i> weekly <i>day-of-week hh:mm</i>}</p> <p>Example:</p> <pre>Device(config)# diagnostic schedule switch 1 test 1-5 on July 3 2013 23:10</pre>	<p>Schedules on-demand diagnostic tests for a specific day and time.</p> <p>The switch <i>number</i> keyword is supported only on stacking switches. The range is from 1 to 8.</p> <p>When specifying the tests to be scheduled, use these options:</p> <ul style="list-style-type: none"> • <i>name</i>—Name of the test that appears in the show diagnostic content command output. • <i>test-id</i>—ID number of the test that appears in the show diagnostic content command output. • <i>test-id-range</i>—ID numbers of the tests that appear in the show diagnostic content command output. • all—All test IDs. • basic—Starts the basic on-demand diagnostic tests. • non-disruptive—Starts the non-disruptive test suite. <p>You can schedule the tests as follows:</p> <ul style="list-style-type: none"> • Daily—Use the daily <i>hh:mm</i> parameter. • Specific day and time—Use the on <i>mm dd yyyy hh:mm</i> parameter. • Weekly—Use the weekly <i>day-of-week hh:mm</i> parameter.

Configuring Health-Monitoring Diagnostics

You can configure health-monitoring diagnostic testing on a Device while it is connected to a live network. You can configure the execution interval for each health-monitoring test, enable the Device to generate a syslog message because of a test failure, and enable a specific test.

Use the **no** form of this command to disable testing.

By default, health monitoring is disabled, but the Device generates a syslog message when a test fails.

Follow these steps to configure and enable the health-monitoring diagnostic tests:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **diagnostic monitor interval switch** *number test* {*name* | *test-id* | *test-id-range* | **all**} *hh:mm:ss milliseconds day*
4. **diagnostic monitor syslog**
5. **diagnostic monitor threshold switch** *number number test* {*name* | *test-id* | *test-id-range* | **all**} **failure count** *count*
6. **diagnostic monitor switch** *number test* {*name* | *test-id* | *test-id-range* | **all**}
7. **end**
8. **show running-config**
9. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	diagnostic monitor interval switch <i>number test</i> { <i>name</i> <i>test-id</i> <i>test-id-range</i> all } <i>hh:mm:ss milliseconds day</i> Example: Device(config)# diagnostic monitor interval switch 2 test 1 12:30:00 750 5	Configures the health-monitoring interval of the specified tests. The switch <i>number</i> keyword is supported only on stacking switches. When specifying the tests, use one of these parameters: <ul style="list-style-type: none"> • <i>name</i>—Name of the test that appears in the show diagnostic content command output. • <i>test-id</i>—ID number of the test that appears in the show diagnostic content command output. • <i>test-id-range</i>—ID numbers of the tests that appear in the show diagnostic content command output. • all—All of the diagnostic tests. When specifying the interval, set these parameters:

	Command or Action	Purpose
		<ul style="list-style-type: none"> • <i>hh:mm:ss</i>—Monitoring interval in hours, minutes, and seconds. The range for <i>hh</i> is 0 to 24, and the range for <i>mm</i> and <i>ss</i> is 0 to 60. • <i>milliseconds</i>—Monitoring interval in milliseconds (ms). The range is from 0 to 999. • <i>day</i>—Monitoring interval in the number of days. The range is from 0 to 20.
Step 4	diagnostic monitor syslog Example: <pre>Device(config)# diagnostic monitor syslog</pre>	(Optional) Configures the switch to generate a syslog message when a health-monitoring test fails.
Step 5	diagnostic monitor threshold switch <i>number number test {name test-id test-id-range all} failure count count</i> Example: <pre>Device(config)# diagnostic monitor threshold switch 2 test 1 failure count 20</pre>	(Optional) Sets the failure threshold for the health-monitoring tests. The switch <i>number</i> keyword is supported only on stacking switches. The range is from 1 to 8. When specifying the tests, use one of these parameters: <ul style="list-style-type: none"> • <i>name</i>—Name of the test that appears in the show diagnostic content command output. • <i>test-id</i>—ID number of the test that appears in the show diagnostic content command output. • <i>test-id-range</i>—ID numbers of the tests that appear in the show diagnostic content command output. • all—All of the diagnostic tests. The range for the failure threshold <i>count</i> is 0 to 99.
Step 6	diagnostic monitor switch <i>number test {name test-id test-id-range all}</i> Example: <pre>Device(config)# diagnostic monitor switch 2 test 1</pre>	Enables the specified health-monitoring tests. The switch <i>number</i> keyword is supported only on stacking switches. The range is from 1 to 8. When specifying the tests, use one of these parameters: <ul style="list-style-type: none"> • <i>name</i>—Name of the test that appears in the show diagnostic content command output. • <i>test-id</i>—ID number of the test that appears in the show diagnostic content command output. • <i>test-id-range</i>—ID numbers of the tests that appear in the show diagnostic content command output. • all—All of the diagnostic tests.

	Command or Action	Purpose
Step 7	end Example: Device(config)# end	Returns to privileged EXEC mode.
Step 8	show running-config Example: Device# show running-config	Verifies your entries.
Step 9	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

What to do next

Use the **no diagnostic monitor interval test***test-id | test-id-range* } global configuration command to change the interval to the default value or to zero. Use the **no diagnostic monitor syslog** command to disable generation of syslog messages when a health-monitoring test fails. Use the **diagnostic monitor threshold test***test-id | test-id-range* }**failure count** command to remove the failure threshold.

Monitoring and Maintaining Online Diagnostics

Displaying Online Diagnostic Tests and Test Results

You can display the online diagnostic tests that are configured for the Device or Device stack and check the test results by using the privileged EXEC **show** commands in this table:

Table 185: Commands for Diagnostic Test Configuration and Results

Command	Purpose
show diagnostic content switch [<i>number</i> all]	Displays the online diagnostics configured for a switch.
show diagnostic status	Displays the currently running diagnostic tests.
show diagnostic result switch [<i>number</i> all] [detail test { <i>name</i> <i>test-id</i> <i>test-id-range</i> all } [detail]]	Displays the online diagnostics test results.
show diagnostic switch [<i>number</i> all] [detail]	Displays the online diagnostics test results.
show diagnostic schedule switch [<i>number</i> all]	Displays the online diagnostics test schedule.

Command	Purpose
<code>show diagnostic post</code>	Displays the POST results. (The output is the same as the <code>show post</code> command output.)

Configuration Examples for Online Diagnostic Tests

Starting Online Diagnostic Tests

After you configure diagnostic tests to run on the switch, use the `diagnostic start` privileged EXEC command to begin diagnostic testing.

After starting the tests, you cannot stop the testing process.

Use this privileged EXEC command to manually start online diagnostic testing.

SUMMARY STEPS

1. `diagnostic start switch number test {name | test-id | test-id-range | all | basic | non-disruptive }`

DETAILED STEPS

	Command or Action	Purpose
Step 1	<p><code>diagnostic start switch number test {name test-id test-id-range all basic non-disruptive }</code></p> <p>Example:</p> <pre>Device# diagnostic start switch 2 test basic</pre>	<p>Starts the diagnostic tests.</p> <p>The <code>switch number</code> keyword is supported only on stacking switches. The range is from 1 to 8.</p> <p>You can specify the tests by using one of these options:</p> <ul style="list-style-type: none"> • <code>name</code>—Enters the name of the test. • <code>test-id</code>—Enters the ID number of the test. • <code>test-id-range</code>—Enters the range of test IDs by using integers separated by a comma and a hyphen. • <code>all</code>—Starts all of the tests. • <code>basic</code>— Starts the basic test suite. • <code>non-disruptive</code>—Starts the non-disruptive test suite.

Example: Configure a Health Monitoring Test

This example shows how to configure a health-monitoring test:

```
Device(config)# diagnostic monitor threshold switch 1 test 1 failure count 50
Device(config)# diagnostic monitor interval switch 1 test TestPortAsicStackPortLoopback
```

Examples: Schedule Diagnostic Test

This example shows how to schedule diagnostic testing for a specific day and time on a specific switch:

```
Device(config)# diagnostic schedule test DiagThermalTest on June 3 2013 22:25
```

This example shows how to schedule diagnostic testing to occur weekly at a certain time on a specific switch:

```
Device(config)# diagnostic schedule switch 1 test 1,2,4-6 weekly saturday 10:30
```

Displaying Online Diagnostics: Examples

This example shows how to display the online diagnostic detailed information on a specific switch:

```
Device# show diagnostic switch 1 detail
```

```
Switch 1: SerialNo :
```

```
Overall Diagnostic Result for Switch 1 : UNTESTED
```

```
Test results: (. = Pass, F = Fail, U = Untested)
```

```
1) TestPortAsicStackPortLoopback ---> U
```

```
Error code -----> 3 (DIAG_SKIPPED)
Total run count -----> 0
Last test testing type -----> n/a
Last test execution time ----> n/a
First test failure time -----> n/a
Last test failure time -----> n/a
Last test pass time -----> n/a
Total failure count -----> 0
Consecutive failure count ---> 0
```

```
2) TestPortAsicLoopback -----> U
```

```
Error code -----> 3 (DIAG_SKIPPED)
Total run count -----> 0
Last test testing type -----> n/a
Last test execution time ----> n/a
First test failure time -----> n/a
Last test failure time -----> n/a
Last test pass time -----> n/a
Total failure count -----> 0
Consecutive failure count ---> 0
```

```
3) TestPortAsicCam -----> U
```

```
Error code -----> 3 (DIAG_SKIPPED)
Total run count -----> 0
Last test testing type -----> n/a
Last test execution time ----> n/a
First test failure time -----> n/a
Last test failure time -----> n/a
Last test pass time -----> n/a
```



```

Total failure count -----> 0
Consecutive failure count ---> 0

-----

4) TestPortAsicMem -----> U

Error code -----> 3 (DIAG_SKIPPED)
Total run count -----> 0
Last test testing type -----> n/a
Last test execution time ----> n/a
First test failure time -----> n/a
Last test failure time -----> n/a
Last test pass time -----> n/a
Total failure count -----> 0
Consecutive failure count ---> 0

-----

5) TestInlinePwrCtrlr -----> U

Error code -----> 3 (DIAG_SKIPPED)
Total run count -----> 0
Last test testing type -----> n/a
Last test execution time ----> n/a
First test failure time -----> n/a
Last test failure time -----> n/a
Last test pass time -----> n/a
Total failure count -----> 0
Consecutive failure count ---> 0

```

This example shows how to display the online diagnostics that are configured on a specific switch:

```
Device# show diagnostic content switch 3
```

```

Switch 1:
Diagnostics test suite attributes:
  B/* - Basic ondemand test / NA
  P/V/* - Per port test / Per device test / NA
  D/N/* - Disruptive test / Non-disruptive test / NA
  S/* - Only applicable to standby unit / NA
  X/* - Not a health monitoring test / NA
  F/* - Fixed monitoring interval test / NA
  E/* - Always enabled monitoring test / NA
  A/I - Monitoring is active / Monitoring is inactive
  R/* - Switch will reload after test list completion / NA
  P/* - will partition stack / NA

```

ID	Test Name	Attributes	Test Interval day hh:mm:ss.ms	Thre- shold
1)	TestPortAsicStackPortLoopback	---> B*N***I**	not configured	n/a
2)	TestPortAsicLoopback	-----> B*D*X**IR*	not configured	n/a
3)	TestPortAsicCam	-----> B*D*X**IR*	not configured	n/a
4)	TestPortAsicRingLoopback	-----> B*D*X**IR*	not configured	n/a
5)	TestMicRingLoopback	-----> B*D*X**IR*	not configured	n/a
6)	TestPortAsicMem	-----> B*D*X**IR*	not configured	n/a

This example shows how to display the online diagnostic results for a switch:

```

Device# show diagnostic result

Switch 1: SerialNo :
Overall diagnostic result: PASS
Test results: (. = Pass, F = Fail, U = Untested)
1) TestPortAsicStackPortLoopback ---> .
2) TestPortAsicLoopback -----> .
3) TestPortAsicCam -----> .
4) TestPortAsicRingLoopback -----> .
5) TestMicRingLoopback -----> .
6) TestPortAsicMem -----> .

```

This example shows how to display the online diagnostic test status:

```

Device# show diagnostic status

<BU> - Bootup Diagnostics, <HM> - Health Monitoring Diagnostics,
<OD> - OnDemand Diagnostics, <SCH> - Scheduled Diagnostics
=====
Card   Description                               Current Running Test           Run by
-----
1      N/A                                          N/A                             N/A
2      TestPortAsicStackPortLoopback             <OD>
      TestPortAsicLoopback                     <OD>
      TestPortAsicCam                           <OD>
      TestPortAsicRingLoopback                 <OD>
      TestMicRingLoopback                      <OD>
      TestPortAsicMem                          <OD>
3      N/A                                          N/A                             N/A
4      N/A                                          N/A                             N/A
=====
Switch#

```

This example shows how to display the online diagnostic test schedule for a switch:

```

Device# show diagnostic schedule switch 1

Current Time = 14:39:49 PST Tue May 5 2013
Diagnostic for Switch 1:
Schedule #1:
To be run daily 12:00
Test ID(s) to be executed: 1.

```

Additional References for Online Diagnostics

Related Documents

Related Topic	Document Title
Online diagnostics commands	<i>Catalyst 2960-XR Switch System Management Command Reference</i>
Platform-independent command references	<i>Cisco IOS 15.3M&T Command References</i>

Related Topic	Document Title
Platform-independent configuration information	<i>Cisco IOS 15.3M&T Configuration Guides</i>

Standards and RFCs

Standard/RFC	Title
None	—

MIBs

MIB	MIBs Link
All supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
<p>The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies.</p> <p>To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds.</p> <p>Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.</p>	http://www.cisco.com/support

Feature History and Information for Configuring Online Diagnostics

Release	Modification
Cisco IOS Release 15.0(2)EX1	This feature was introduced.



CHAPTER 97

Troubleshooting the Software Configuration

This chapter describes how to identify and resolve software problems related to the Cisco IOS software on the switch. Depending on the nature of the problem, you can use the command-line interface (CLI), Device Manager, or Network Assistant to identify and solve problems.

Additional troubleshooting information, such as LED descriptions, is provided in the hardware installation guide.

- [Information About Troubleshooting the Software Configuration, on page 1939](#)
- [How to Troubleshoot the Software Configuration, on page 1945](#)
- [Verifying Troubleshooting of the Software Configuration, on page 1959](#)
- [Scenarios for Troubleshooting the Software Configuration, on page 1963](#)
- [Configuration Examples for Troubleshooting Software, on page 1965](#)
- [Additional References for Troubleshooting Software Configuration, on page 1967](#)
- [Feature History and Information for Troubleshooting Software Configuration, on page 1968](#)

Information About Troubleshooting the Software Configuration

Software Failure on a Switch

Switch software can be corrupted during an upgrade by downloading the incorrect file to the switch, and by deleting the image file. In all of these cases, the switch does not pass the power-on self-test (POST), and there is no connectivity.

Lost or Forgotten Password on a Device

The default configuration for the device allows an end user with physical access to the device to recover from a lost password by interrupting the boot process during power-on and by entering a new password. These recovery procedures require that you have physical access to the device.



Note On these devices, a system administrator can disable some of the functionality of this feature by allowing an end user to reset a password only by agreeing to return to the default configuration. If you are an end user trying to reset a password when password recovery has been disabled, a status message reminds you to return to the default configuration during the recovery process.



Note You cannot recover encryption password key, when Cisco WLC configuration is copied from one Cisco WLC to another (in case of an RMA).

Power over Ethernet Ports

A Power over Ethernet (PoE) switch port automatically supplies power to one of these connected devices if the switch detects that there is no power on the circuit:

- a Cisco pre-standard powered device (such as a Cisco IP Phone or a Cisco Aironet Access Point)
- an IEEE 802.3af-compliant powered device
- an IEEE 802.3at-compliant powered device

A powered device can receive redundant power when it is connected to a PoE switch port and to an AC power source. The device does not receive redundant power when it is only connected to the PoE port.

After the switch detects a powered device, the switch determines the device power requirements and then grants or denies power to the device. The switch can also detect the real-time power consumption of the device by monitoring and policing the power usage.

For more information, see the "Configuring PoE" chapter in the *Catalyst 2960-XR Switch Interface and Hardware Component Configuration Guide*.

Disabled Port Caused by Power Loss

If a powered device (such as a Cisco IP Phone 7910) that is connected to a PoE Device port and powered by an AC power source loses power from the AC power source, the device might enter an error-disabled state. To recover from an error-disabled state, enter the **shutdown** interface configuration command, and then enter the **no shutdown** interface command. You can also configure automatic recovery on the Device to recover from the error-disabled state.

On a Device, the **errdisable recovery cause loopback** and the **errdisable recovery interval seconds** global configuration commands automatically take the interface out of the error-disabled state after the specified period of time.

Monitoring PoE Port Status

- **show controllers power inline** privileged EXEC command
- **show power inline** EXEC command
- **debug ilpower** privileged EXEC command

Disabled Port Caused by False Link-Up

If a Cisco powered device is connected to a port and you configure the port by using the **power inline never** interface configuration command, a false link-up can occur, placing the port into an error-disabled state. To take the port out of the error-disabled state, enter the **shutdown** and the **no shutdown** interface configuration commands.

You should not connect a Cisco powered device to a port that has been configured with the **power inline never** command.

Ping

The Device supports IP ping, which you can use to test connectivity to remote hosts. Ping sends an echo request packet to an address and waits for a reply. Ping returns one of these responses:

- Normal response—The normal response (*hostname is alive*) occurs in 1 to 10 seconds, depending on network traffic.
- Destination does not respond—If the host does not respond, a *no-answer* message is returned.
- Unknown host—If the host does not exist, an *unknown host* message is returned.
- Destination unreachable—If the default gateway cannot reach the specified network, a *destination-unreachable* message is returned.
- Network or host unreachable—If there is no entry in the route table for the host or network, a *network or host unreachable* message is returned.

Layer 2 Traceroute

The Layer 2 traceroute feature allows the switch to identify the physical path that a packet takes from a source device to a destination device. Layer 2 traceroute supports only unicast source and destination MAC addresses. Traceroute finds the path by using the MAC address tables of the Device in the path. When the Device detects a device in the path that does not support Layer 2 traceroute, the Device continues to send Layer 2 trace queries and lets them time out.

The Device can only identify the path from the source device to the destination device. It cannot identify the path that a packet takes from source host to the source device or from the destination device to the destination host.

Layer 2 Traceroute Guidelines

- Cisco Discovery Protocol (CDP) must be enabled on all the devices in the network. For Layer 2 traceroute to function properly, do not disable CDP.
If any devices in the physical path are transparent to CDP, the switch cannot identify the path through these devices.
- A Device is reachable from another Device when you can test connectivity by using the **ping** privileged EXEC command. All Device in the physical path must be reachable from each other.
- The maximum number of hops identified in the path is ten.
- You can enter the **traceroute mac** or the **traceroute mac ip** privileged EXEC command on a Device that is not in the physical path from the source device to the destination device. All Device in the path must be reachable from this switch.
- The **traceroute mac** command output shows the Layer 2 path only when the specified source and destination MAC addresses belong to the same VLAN. If you specify source and destination MAC addresses that belong to different VLANs, the Layer 2 path is not identified, and an error message appears.

- If you specify a multicast source or destination MAC address, the path is not identified, and an error message appears.
- If the source or destination MAC address belongs to multiple VLANs, you must specify the VLAN to which both the source and destination MAC addresses belong. If the VLAN is not specified, the path is not identified, and an error message appears.
- The **traceroute mac ip** command output shows the Layer 2 path when the specified source and destination IP addresses belong to the same subnet. When you specify the IP addresses, the Device uses the Address Resolution Protocol (ARP) to associate the IP addresses with the corresponding MAC addresses and the VLAN IDs.
 - If an ARP entry exists for the specified IP address, the Device uses the associated MAC address and identifies the physical path.
 - If an ARP entry does not exist, the Device sends an ARP query and tries to resolve the IP address. If the IP address is not resolved, the path is not identified, and an error message appears.
- When multiple devices are attached to one port through hubs (for example, multiple CDP neighbors are detected on a port), the Layer 2 traceroute feature is not supported. When more than one CDP neighbor is detected on a port, the Layer 2 path is not identified, and an error message appears.
- This feature is not supported in Token Ring VLANs.

IP Traceroute

You can use IP traceroute to identify the path that packets take through the network on a hop-by-hop basis. The command output displays all network layer (Layer 3) devices, such as routers, that the traffic passes through on the way to the destination.

Your Device can participate as the source or destination of the **traceroute** privileged EXEC command and might or might not appear as a hop in the **traceroute** command output. If the Device is the destination of the traceroute, it is displayed as the final destination in the traceroute output. Intermediate Device do not show up in the traceroute output if they are only bridging the packet from one port to another within the same VLAN. However, if the intermediate Device is a multilayer Device that is routing a particular packet, this Device shows up as a hop in the traceroute output.

The **traceroute** privileged EXEC command uses the Time To Live (TTL) field in the IP header to cause routers and servers to generate specific return messages. Traceroute starts by sending a User Datagram Protocol (UDP) datagram to the destination host with the TTL field set to 1. If a router finds a TTL value of 1 or 0, it drops the datagram and sends an Internet Control Message Protocol (ICMP) time-to-live-exceeded message to the sender. Traceroute finds the address of the first hop by examining the source address field of the ICMP time-to-live-exceeded message.

To identify the next hop, traceroute sends a UDP packet with a TTL value of 2. The first router decrements the TTL field by 1 and sends the datagram to the next router. The second router sees a TTL value of 1, discards the datagram, and returns the time-to-live-exceeded message to the source. This process continues until the TTL is incremented to a value large enough for the datagram to reach the destination host (or until the maximum TTL is reached).

To learn when a datagram reaches its destination, traceroute sets the UDP destination port number in the datagram to a very large value that the destination host is unlikely to be using. When a host receives a datagram destined to itself containing a destination port number that is unused locally, it sends an ICMP *port-unreachable*

error to the source. Because all errors except port-unreachable errors come from intermediate hops, the receipt of a port-unreachable error means that this message was sent by the destination port.

Time Domain Reflector Guidelines

You can use the Time Domain Reflector (TDR) feature to diagnose and resolve cabling problems. When running TDR, a local device sends a signal through a cable and compares the reflected signal to the initial signal.

TDR is supported only on 10/100/1000 copper Ethernet ports. It is not supported on 10-Gigabit Ethernet ports and on SFP module ports.

TDR can detect these cabling problems:

- Open, broken, or cut twisted-pair wires—The wires are not connected to the wires from the remote device.
- Shorted twisted-pair wires—The wires are touching each other or the wires from the remote device. For example, a shorted twisted pair can occur if one wire of the twisted pair is soldered to the other wire.

If one of the twisted-pair wires is open, TDR can find the length at which the wire is open.

Use TDR to diagnose and resolve cabling problems in these situations:

- Replacing a Device
- Setting up a wiring closet
- Troubleshooting a connection between two devices when a link cannot be established or when it is not operating properly

When you run TDR, the Device reports accurate information in these situations:

- The cable for the gigabit link is a solid-core cable.
- The open-ended cable is not terminated.

When you run TDR, the Device does not report accurate information in these situations:

- The cable for the gigabit link is a twisted-pair cable or is in series with a solid-core cable.
- The link is a 10-megabit or a 100-megabit link.
- The cable is a stranded cable.
- The link partner is a Cisco IP Phone.
- The link partner is not IEEE 802.3 compliant.

Debug Commands



Caution Because debugging output is assigned high priority in the CPU process, it can render the system unusable. For this reason, use **debug** commands only to troubleshoot specific problems or during troubleshooting sessions with Cisco technical support staff. It is best to use **debug** commands during periods of lower network traffic and fewer users. Debugging during these periods decreases the likelihood that increased **debug** command processing overhead will affect system use.

All **debug** commands are entered in privileged EXEC mode, and most **debug** commands take no arguments.

Onboard Failure Logging on the Switch

You can use the onboard failure logging (OBFL) feature to collect information about the Device. The information includes uptime, temperature, and voltage information and helps Cisco technical support representatives to troubleshoot Device problems. We recommend that you keep OBFL enabled and do not erase the data stored in the flash memory.

By default, OBFL is enabled. It collects information about the Device and small form-factor pluggable (SFP) modules. The Device stores this information in the flash memory:

- CLI commands—Record of the OBFL CLI commands that are entered on a standalone Device or a switch stack member.
- Environment data—Unique device identifier (UDI) information for a standalone Device or a switch stack member and for all the connected FRU devices: the product identification (PID), the version identification (VID), and the serial number.
- Message—Record of the hardware-related system messages generated by a standalone Device or a switch stack member.
- Power over Ethernet (PoE)—Record of the power consumption of PoE ports on a standalone Device or a switch stack member.
- Temperature—Temperature of a standalone Device or a switch stack member.
- Uptime data—Time when a standalone Device or a switch stack member starts, the reason the Device restarts, and the length of time the Device has been running since it last restarted.
- Voltage—System voltages of a standalone Device or a switch stack member.

You should manually set the system clock or configure it by using Network Time Protocol (NTP).

When the Device is running, you can retrieve the OBFL data by using the **show logging onboard** privileged EXEC commands. If the Device fails, contact your Cisco technical support representative to find out how to retrieve the data.

When an OBFL-enabled Device is restarted, there is a 10-minute delay before logging of new data begins.

Possible Symptoms of High CPU Utilization

Excessive CPU utilization might result in these symptoms, but the symptoms might also result from other causes:



Note You may see increased system memory usage when Cisco Catalyst 4500E Supervisor Engine 8-E is used in wireless mode.

- Spanning tree topology changes
- EtherChannel links brought down due to loss of communication
- Failure to respond to management requests (ICMP ping, SNMP timeouts, slow Telnet or SSH sessions)
- UDLD flapping
- IP SLAs failures because of SLAs responses beyond an acceptable threshold
- DHCP or IEEE 802.1x failures if the switch does not forward or respond to requests

Layer 3 switches:

- Dropped packets or increased latency for packets routed in software
- BGP or OSPF routing topology changes
- HSRP flapping

How to Troubleshoot the Software Configuration

Recovering from a Software Failure

Switch software can be corrupted during an upgrade by downloading the wrong file to the switch, and by deleting the image file. In all of these cases, the switch does not pass the power-on self-test (POST), and there is no connectivity.

This procedure uses the Xmodem Protocol to recover from a corrupt or wrong image file. There are many software packages that support the Xmodem Protocol, and this procedure is largely dependent on the emulation software that you are using.

This recovery procedure requires that you have physical access to the switch.

Step 1 From your PC, download the software image tar file (*image_filename.tar*) from Cisco.com. The Cisco IOS image is stored as a bin file in a directory in the tar file. For information about locating the software image files on Cisco.com, see the release notes.

Step 2 Extract the bin file from the tar file. If you are using Windows, use a zip program that can read a tar file. Use the zip program to navigate. If you are using Windows, use a zip program that can read a tar file. Use the zip program to navigate. If you are using UNIX, follow these steps:

- a) Display the contents of the tar file by using the **tar -tvf <image_filename.tar>** UNIX command.

Example:

```
unix-1% tar -tvf image_filename.tar
```

- b) Locate the bin file, and extract it by using the `tar -xvf <image_filename.tar> <image_filename.bin>` UNIX command.

Example:

```
unix-1% tar -xvf image_filename.tar image_filename.bin
x c2960x-universalk9-mz-150-2.EX1/c2960x-universalk9-mz-150-2.EX1.bin, 2928176 bytes, 5720
tape blocks
```

- c) Verify that the bin file was extracted by using the `ls -l <image_filename.bin>` UNIX command.

Example:

```
unix-1% ls -l image_filename.bin
-rw-r--r--  1 boba      2928176 Apr 21 12:01
c2960x-universalk9-mz.150-2.0.66.UCP/c2960x-universalk9-mz.150-2.0.66.UCP.bin
```

- Step 3** Connect your PC with terminal-emulation software supporting the Xmodem Protocol to the switch console port.
- Step 4** Set the line speed on the emulation software to 9600 baud.
- Step 5** Unplug the switch power cord.
- Step 6** Press the **Mode** button, and at the same time reconnect the power cord to the switch. You can release the Mode button a second or two after the LED above port 1 goes off. Several lines of information about the software appear along with instructions.

Example:

The system has been interrupted prior to initializing the flash file system. The following commands will initialize the flash file system, and finish loading the operating system

```
software#
```

```
flash_init
load_helper
boot
```

- Step 7** Initialize the flash file system.

Example:

```
switch: flash_init
```

- Step 8** If you had set the console port speed to any speed other than 9600, it has been reset to that particular speed. Change the emulation software line speed to match that of the switch console port.

- Step 9** Load any helper files.

Example:

```
switch: load_helper
```

- Step 10** Start the file transfer by using the Xmodem Protocol.

Example:

```
switch: copy xmodem: flash:image_filename.bin
```

- Step 11** After the Xmodem request appears, use the appropriate command on the terminal-emulation software to start the transfer and to copy the software image into flash memory.
- Step 12** Boot the newly downloaded Cisco IOS image.
- Example:**
- ```
switch: boot flash:image_filename.bin
```
- Step 13** Use the **archive download-sw** privileged EXEC command to download the software image to the switch or to the switch stack.
- Step 14** Use the **reload** privileged EXEC command to restart the switch and to verify that the new software image is operating properly.
- Step 15** Delete the **flash:image\_filename.bin** file from the switch.
- 

## Recovering from a Lost or Forgotten Password

The default configuration for the switch allows an end user with physical access to the switch to recover from a lost password by interrupting the boot process during power-on and by entering a new password. These recovery procedures require that you have physical access to the switch.



**Note** On these switches, a system administrator can disable some of the functionality of this feature by allowing an end user to reset a password only by agreeing to return to the default configuration. If you are an end user trying to reset a password when password recovery has been disabled, a status message shows this during the recovery process.

---

You enable or disable password recovery by using the **service password-recovery** global configuration command.

The switch supports homogeneous stacking, but does not support mixed stacking.

---

- Step 1** Connect a terminal or PC to the switch.
- Connect a terminal or a PC with terminal-emulation software to the switch console port.
- Or
- Connect a PC to the Ethernet management port.
- Step 2** Set the line speed on the emulation software to 9600 baud.
- Step 3** On a switch, power off the switch.
- Step 4** Reconnect the power cord to the switch. Within 15 seconds, press the **Mode** button while the System LED is still flashing green. Continue pressing the **Mode** button until all the system LEDs turn on and remain solid, then release the **Mode** button.

Several lines of information about the software appear with instructions, informing you if the password recovery procedure has been disabled or not.

- If you see a message that begins with this statement:

The system has been interrupted prior to initializing the flash file system. The following commands will initialize the flash file system

proceed to the "Procedure with Password Recovery Enabled" section, and follow the steps.

- If you see a message that begins with this statement:

The password-recovery mechanism has been triggered, but is currently disabled.

proceed to the "Procedure with Password Recovery Disabled" section, and follow the steps.

**Step 5** After recovering the password, reload the switch.

On a switch:

```
Switch> reload
Proceed with reload? [confirm] y
```

---

## Procedure with Password Recovery Enabled

If the password-recovery operation is enabled, this message appears:

The system has been interrupted prior to initializing the flash file system. The following commands will initialize the flash file system, and finish loading the operating system software:

```
flash_init
load_helper
boot
```

---

**Step 1** Initialize the flash file system.

Device: **flash\_init**

**Step 2** If you had set the console port speed to any number other than 9600, it has been reset to that particular speed. Change the emulation software line speed to match that of the switch console port.

**Step 3** Load any helper files.

Device: **load\_helper**

**Step 4** Display the contents of flash memory.

```
Device: dir: flash:
Directory of flash:
 13 drwx 192 Mar 01 2013 22:30:48
c2960x-universalk9-mz-150-2.EX1/c2960x-universalk9-mz-150-2.EX1.bin
 11 -rwx 5825 Mar 01 2013 22:31:59 config.text

16128000 bytes total (10003456 bytes free)
```

**Step 5** Rename the configuration file to config.text.old

This file contains the password definition.

```
Device: rename flash: config.text flash: config.text.old
```

**Step 6** Boot up the system.

```
Device: boot
```

You are prompted to start the setup program. Enter **N** at the prompt.

```
Continue with the configuration dialog?? [yes/no]: No
```

**Step 7** At the switch prompt, enter privileged EXEC mode.

```
Device> enable
Switch#
```

**Step 8** Rename the configuration file to its original name.

```
Device# rename flash: config.text.old flash: config.text
```

**Note** Before continuing to Step 9, power on any connected stack members and wait until they have completely initialized. Failure to follow this step can result in a lost configuration depending on how your device is set up.

**Step 9** Copy the configuration file into memory

```
Device# copy flash: config.text system: running-config
Source filename [config.text]?
Destination filename [running-config]?
```

Press **Return** in response to the confirmation prompts. The configuration file is now reloaded, and you can change the password.

**Step 10** Enter global configuration mode.

```
Device# configure terminal
```

**Step 11** Change the password.

```
Device(config)# enable secret password
```

The secret password can be from 1 to 25 alphanumeric characters, can start with a number, is case sensitive, and allows spaces but ignores leading spaces.

**Step 12** Return to privileged EXEC mode.

```
Device(config)# exit
Switch#
```

**Step 13** Write the running configuration to the startup configuration file.

```
Device# copy running-config startup-config
```

The new password is now in the startup configuration.

**Note** This procedure is likely to leave your switch virtual interface in a shutdown state. You can see which interface is in this state by entering the **show running-config** privileged EXEC command. To reenabte the interface, enter the **interface vlan *vlan-id*** global configuration command, and specify the VLAN ID of the shutdown interface. With the switch in interface configuration mode, enter the **no shutdown** command.

**Step 14** Boot the device with the *packages.conf* file from flash.

```
Device: boot flash:packages.conf
```

**Step 15** Reload the switch stack.

```
Device# reload
```

---

## Procedure with Password Recovery Disabled

If the password-recovery mechanism is disabled, this message appears:

```
The password-recovery mechanism has been triggered, but
is currently disabled. Access to the boot loader prompt
through the password-recovery mechanism is disallowed at
this point. However, if you agree to let the system be
reset back to the default system configuration, access
to the boot loader prompt can still be allowed.
```

```
Would you like to reset the system back to the default configuration (y/n)?
```




---

**Caution** Returning the Device to the default configuration results in the loss of all existing configurations. We recommend that you contact your system administrator to verify if there are backup Device and VLAN configuration files.

---

- If you enter **n** (no), the normal boot process continues as if the **Mode** button had not been pressed; you cannot access the boot loader prompt, and you cannot enter a new password. You see the message:

```
Press Enter to continue.....
```

- If you enter **y** (yes), the configuration file in flash memory and the VLAN database file are deleted. When the default configuration loads, you can reset the password.
- 

**Step 1** Choose to continue with password recovery and delete the existing configuration:

```
Would you like to reset the system back to the default configuration (y/n)? y
```

**Step 2** Display the contents of flash memory:



```
Device: dir flash:
```

The Device file system appears.

**Step 3** Boot up the system:

```
Device: boot
```

You are prompted to start the setup program. To continue with password recovery, enter **N** at the prompt:

```
Continue with the configuration dialog? [yes/no]: N
```

**Step 4** At the Device prompt, enter privileged EXEC mode:

```
Device> enable
```

**Step 5** Enter global configuration mode:

```
Device# configure terminal
```

**Step 6** Change the password:

```
Device(config)# enable secret password
```

The secret password can be from 1 to 25 alphanumeric characters, can start with a number, is case sensitive, and allows spaces but ignores leading spaces.

**Step 7** Return to privileged EXEC mode:

```
Device(config)# exit
Device#
```

**Note** Before continuing to Step 9, power on any connected stack members and wait until they have completely initialized.

**Step 8** Write the running configuration to the startup configuration file:

```
Device# copy running-config startup-config
```

The new password is now in the startup configuration.

**Step 9** You must now reconfigure the Device. If the system administrator has the backup Device and VLAN configuration files available, you should use those.

---

## Recovering from a Command Switch Failure

This section describes how to recover from a failed command switch. You can configure a redundant command switch group by using the Hot Standby Router Protocol (HSRP).




---

**Note** HSRP is the preferred method for supplying redundancy to a cluster.

---




---

**Note** This feature is introduced from Cisco IOS Release 15.2(5)E2.

---

If you have not configured a standby command switch, and your command switch loses power or fails in some other way, management contact with the member switches is lost, and you must install a new command switch. However, connectivity between switches that are still connected is not affected, and the member switches forward packets as usual. You can manage the members as standalone switches through the console port, or, if they have IP addresses, through the other management interfaces.

You can prepare for a command switch failure by assigning an IP address to a member switch or another switch that is command-capable, making a note of the command-switch password, and cabling your cluster to provide redundant connectivity between the member switches and the replacement command switch. These sections describe two solutions for replacing a failed command switch:

- Replacing a Failed Command Switch with a Cluster Member
- Replacing a Failed Command Switch with Another Switch

These recovery procedures require that you have physical access to the switch. For information on command-capable switches, see the release notes.

## Replacing a Failed Command Switch with a Cluster Member

To replace a failed command switch with a command-capable member in the same cluster, follow these steps

- 
- Step 1** Disconnect the command switch from the member switches, and physically remove it from the cluster.
- Step 2** Insert the member switch in place of the failed command switch, and duplicate its connections to the cluster members.
- Step 3** Start a CLI session on the new command switch.
- You can access the CLI by using the console port or, if an IP address has been assigned to the switch, by using Telnet. For details about using the console port, see *Catalyst 2960-X Switch Hardware Installation Guide*.
- Step 4** At the switch prompt, enter privileged EXEC mode.
- Example:**
- ```
Switch> enable
Switch#
```
- Step 5** Enter the password of the *failed command switch*.
- Step 6** Enter global configuration mode.
- Example:**
- ```
Switch# configure terminal
```
- Enter configuration commands, one per line. End with CNTL/Z.
- Step 7** Remove the member switch from the cluster.

**Example:**

```
Switch(config)# no cluster commander-address
```

**Step 8** Return to privileged EXEC mode.

**Example:**

```
Switch(config)# end
Switch#
```

**Step 9** Use the setup program to configure the switch IP information. This program prompts you for IP address information and passwords. From privileged EXEC mode, enter EXEC mode, enter **setup**, and press **Return**.

**Example:**

```
Switch# setup

--- System Configuration Dialog ---
Continue with configuration dialog? [yes/no]: y
At any point you may enter a question mark '?' for help.
Use ctrl-c to abort configuration dialog at any prompt.
Default settings are in square brackets '['].
Basic management setup configures only enough connectivity
for management of the system, extended setup will ask you
to configure each interface on the system
Would you like to enter basic management setup? [yes/no]:
```

**Step 10** Enter **Y** at the first prompt.

**Example:**

The prompts in the setup program vary depending on the member switch that you selected to be the command switch:

```
Continue with configuration dialog? [yes/no]: y
```

or

```
Configuring global parameters:
```

If this prompt does not appear, enter **enable**, and press **Return**. Enter **setup**, and press **Return** to start the setup program.

**Step 11** Respond to the questions in the setup program.

When prompted for the hostname, it is limited to 28 characters and 31 characters on a member switch. Do not use *-n*, where *n* is a number, as the last characters in a hostname for any switch. When prompted for the Telnet (virtual terminal) password, it is 1 to 25 alphanumeric characters, is case sensitive, allows spaces, but ignores leading spaces.

**Step 12** When prompted for the **enable secret** and **enable** passwords, enter the passwords of the *failed command switch* again.

**Step 13** When prompted, make sure to enable the switch as the cluster command switch, and press **Return**.

**Step 14** When prompted, assign a name to the cluster, and press **Return**.

The cluster name can be 1 to 31 alphanumeric characters, dashes, or underscores.

**Step 15** After the initial configuration displays, verify that the addresses are correct.

**Step 16** If the displayed information is correct, enter **Y**, and press **Return**.

If this information is not correct, enter **N**, press **Return**, and begin again at Step 9.

- Step 17** Start your browser, and enter the IP address of the new command switch.
- Step 18** From the Cluster menu, select **Add to Cluster** to display a list of candidate switches to add to the cluster.

## Replacing a Failed Command Switch with Another Switch

To replace a failed command switch with a switch that is command-capable but not part of the cluster, follow these steps:

- Step 1** Insert the new switch in place of the failed command switch, and duplicate its connections to the cluster members.
- Step 2** You can access the CLI by using the console port or, if an IP address has been assigned to the switch, by using Telnet. For details about using the console port, see the switch hardware installation guide.
- Step 3** At the switch prompt, enter privileged EXEC mode.

**Example:**

```
Switch> enable
Switch#
```

- Step 4** Enter the password of the *failed command switch*.
- Step 5** Use the setup program to configure the switch IP information. This program prompts you for IP address information and passwords. From privileged EXEC mode, enter EXEC mode, enter **setup**, and press **Return**.

**Example:**

```
Switch# setup

--- System Configuration Dialog ---
Continue with configuration dialog? [yes/no]: y
At any point you may enter a question mark '?' for help.
Use ctrl-c to abort configuration dialog at any prompt.
Default settings are in square brackets '[]'.
Basic management setup configures only enough connectivity
for management of the system, extended setup will ask you
to configure each interface on the system
Would you like to enter basic management setup? [yes/no]:
```

- Step 6** Enter **Y** at the first prompt.

**Example:**

The prompts in the setup program vary depending on the member switch that you selected to be the command switch:

```
Continue with configuration dialog? [yes/no]: y
```

or

```
Configuring global parameters:
```

If this prompt does not appear, enter **enable**, and press **Return**. Enter **setup**, and press **Return** to start the setup program.

- Step 7** Respond to the questions in the setup program.

When prompted for the hostname, it is limited to 28 characters and 31 characters on a member switch. Do not use *-n*, where *n* is a number, as the last characters in a hostname for any switch. When prompted for the Telnet (virtual terminal) password, it is 1 to 25 alphanumeric characters, is case sensitive, allows spaces, but ignores leading spaces.

**Step 8** When prompted for the **enable secret** and **enable** passwords, enter the passwords of the *failed command switch* again.

**Step 9** When prompted, make sure to enable the switch as the cluster command switch, and press **Return**.

**Step 10** When prompted, assign a name to the cluster, and press **Return**.

The cluster name can be 1 to 31 alphanumeric characters, dashes, or underscores.

**Step 11** After the initial configuration displays, verify that the addresses are correct.

**Step 12** If the displayed information is correct, enter **Y**, and press **Return**.

If this information is not correct, enter **N**, press **Return**, and begin again at Step 9.

**Step 13** Start your browser, and enter the IP address of the new command switch.

**Step 14** From the Cluster menu, select **Add to Cluster** to display a list of candidate switches to add to the cluster.

---

## Preventing Switch Stack Problems

To prevent switch stack problems, you should do the following:

- Make sure that the Device that you add to or remove from the switch stack are powered off. For all powering considerations in switch stacks, see the “Switch Installation” chapter in the hardware installation guide.
- Press the **Mode** button on a stack member until the Stack mode LED is on. The last two port LEDs on the Device should be green. Depending on the Device model, the last two ports are either 10/100/1000 ports or small form-factor pluggable (SFP) module. If one or both of the last two port LEDs are not green, the stack is not operating at full bandwidth.
- We recommend using only one CLI session when managing the switch stack. Be careful when using multiple CLI sessions to the active stack. Commands that you enter in one session are not displayed in the other sessions. Therefore, it is possible that you might not be able to identify the session from which you entered a command.
- Manually assigning stack member numbers according to the placement of the Device in the stack can make it easier to remotely troubleshoot the switch stack. However, you need to remember that the Device have manually assigned numbers if you add, remove, or rearrange Device later. Use the **switch** *current-stack-member-number* **renumber** *new-stack-member-number* global configuration command to manually assign a stack member number.

If you replace a stack member with an identical model, the new Device functions with the exact same configuration as the replaced Device. This is also assuming the new Device is using the same member number as the replaced Device.

Removing powered-on stack members causes the switch stack to divide (partition) into two or more switch stacks, each with the same configuration. If you want the switch stacks to remain separate, change the IP address or addresses of the newly created switch stacks. To recover from a partitioned switch stack, follow these steps:

1. Power off the newly created switch stacks.
2. Reconnect them to the original switch stack through their StackWise Plus ports.

### 3. Power on the Device.

For the commands that you can use to monitor the switch stack and its members, see the *Displaying Switch Stack Information* section.

## Preventing Autonegotiation Mismatches

The IEEE 802.3ab autonegotiation protocol manages the Device settings for speed (10 Mb/s, 100 Mb/s, and 1000 Mb/s, excluding SFP module ports) and duplex (half or full). There are situations when this protocol can incorrectly align these settings, reducing performance. A mismatch occurs under these circumstances:

- A manually set speed or duplex parameter is different from the manually set speed or duplex parameter on the connected port.
- A port is set to autonegotiate, and the connected port is set to full duplex with no autonegotiation.

To maximize Device performance and ensure a link, follow one of these guidelines when changing the settings for duplex and speed:

- Let both ports autonegotiate both speed and duplex.
- Manually set the speed and duplex parameters for the ports on both ends of the connection.




---

**Note** If a remote device does not autonegotiate, configure the duplex settings on the two ports to match. The speed parameter can adjust itself even if the connected port does not autonegotiate.

---

## Troubleshooting SFP Module Security and Identification

Cisco small form-factor pluggable (SFP) modules have a serial EEPROM that contains the module serial number, the vendor name and ID, a unique security code, and cyclic redundancy check (CRC). When an SFP module is inserted in the Device, the Device software reads the EEPROM to verify the serial number, vendor name and vendor ID, and recompute the security code and CRC. If the serial number, the vendor name or vendor ID, the security code, or CRC is invalid, the software generates a security error message and places the interface in an error-disabled state.




---

**Note** The security error message references the GBIC\_SECURITY facility. The Device supports SFP modules and does not support GBIC modules. Although the error message text refers to GBIC interfaces and modules, the security messages actually refer to the SFP modules and module interfaces.

---

If you are using a non-Cisco SFP module, remove the SFP module from the Device, and replace it with a Cisco module. After inserting a Cisco SFP module, use the **errdisable recovery cause gbic-invalid** global configuration command to verify the port status, and enter a time interval for recovering from the error-disabled state. After the elapsed interval, the Device brings the interface out of the error-disabled state and retries the operation. For more information about the **errdisable recovery** command, see the command reference for this release.

If the module is identified as a Cisco SFP module, but the system is unable to read vendor-data information to verify its accuracy, an SFP module error message is generated. In this case, you should remove and reinsert the SFP module. If it continues to fail, the SFP module might be defective.

## Monitoring SFP Module Status

You can check the physical or operational status of an SFP module by using the **show interfaces transceiver** privileged EXEC command. This command shows the operational status, such as the temperature and the current for an SFP module on a specific interface and the alarm status. You can also use the command to check the speed and the duplex settings on an SFP module. For more information, see the **show interfaces transceiver** command in the command reference for this release.

## Executing Ping

If you attempt to ping a host in a different IP subnetwork, you must define a static route to the network or have IP routing configured to route between those subnets.

IP routing is disabled by default on all Device.



**Note** Though other protocol keywords are available with the **ping** command, they are not supported in this release.

Use this command to ping another device on the network from the Device:

| Command                                                                                | Purpose                                                                         |
|----------------------------------------------------------------------------------------|---------------------------------------------------------------------------------|
| <p><b>ping ip</b> <i>host</i>   <i>address</i></p> <pre>Device# ping 172.20.52.3</pre> | Pings a remote host through IP or by supplying the hostname or network address. |

## Monitoring Temperature

The Device monitors the temperature conditions and uses the temperature information to control the fans.

Use the **show env temperature status** privileged EXEC command to display the temperature value, state, and thresholds. The temperature value is the temperature in the Device (not the external temperature). You can configure only the yellow threshold level (in Celsius) by using the **system env temperature threshold yellow** *value* global configuration command to set the difference between the yellow and red thresholds. You cannot configure the green or red thresholds. For more information, see the command reference for this release.

## Monitoring the Physical Path

You can monitor the physical path that a packet takes from a source device to a destination device by using one of these privileged EXEC commands:

Table 186: Monitoring the Physical Path

| Command                                                                                                                                               | Purpose                                                                                                                                              |
|-------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------|
| <code>tracetroute mac [interface interface-id] {source-mac-address} [interface interface-id] {destination-mac-address} [vlan vlan-id] [detail]</code> | Displays the Layer 2 path taken by the packets from the specified source MAC address to the specified destination MAC address.                       |
| <code>tracetroute mac ip {source-ip-address   source-hostname} {destination-ip-address   destination-hostname} [detail]</code>                        | Displays the Layer 2 path taken by the packets from the specified source IP address or hostname to the specified destination IP address or hostname. |

## Executing IP Traceroute



**Note** Though other protocol keywords are available with the `tracetroute` privileged EXEC command, they are not supported in this release.

| Command                                                                              | Purpose                                                |
|--------------------------------------------------------------------------------------|--------------------------------------------------------|
| <code>tracetroute ip host</code><br>Device# <code>tracetroute ip 192.51.100.1</code> | Traces the path that packets take through the network. |

## Running TDR and Displaying the Results

To run TDR, enter the `test cable-diagnostics tdr interface interface-id` privileged EXEC command.

To display the results, enter the `show cable-diagnostics tdr interface interface-id` privileged EXEC command.

## Redirecting Debug and Error Message Output

By default, the network server sends the output from `debug` commands and system error messages to the console. If you use this default, you can use a virtual terminal connection to monitor debug output instead of connecting to the console port or the Ethernet management port.

Possible destinations include the console, virtual terminals, internal buffer, and UNIX hosts running a syslog server. The syslog format is compatible with 4.3 Berkeley Standard Distribution (BSD) UNIX and its derivatives.



**Note** Be aware that the debugging destination you use affects system overhead. When you log messages to the console, very high overhead occurs. When you log messages to a virtual terminal, less overhead occurs. Logging messages to a syslog server produces even less, and logging to an internal buffer produces the least overhead of any method.

For more information about system message logging, see *Configuring System Message Logging*.



## Using the show platform forward Command

The output from the **show platform forward** privileged EXEC command provides some useful information about the forwarding results if a packet entering an interface is sent through the system. Depending upon the parameters entered about the packet, the output provides lookup table results and port maps used to calculate forwarding destinations, bitmaps, and egress information.

Most of the information in the output from the command is useful mainly for technical support personnel, who have access to detailed information about the Device application-specific integrated circuits (ASICs). However, packet forwarding information can also be helpful in troubleshooting.

## Configuring OBFL



**Caution** We recommend that you do not disable OBFL and that you do not remove the data stored in the flash memory.

- To enable OBFL, use the **hw-switch switch [switch-number] logging onboard [message level level]** global configuration command. On switches, the range for *switch-number* is from 1 to 9. Use the **message level level** parameter to specify the severity of the hardware-related messages that the switch generates and stores in the flash memory.
- To copy the OBFL data to the local network or a specific file system, use the **copy onboard switch switch-number url url-destination** privileged EXEC command.
- To disable OBFL, use the **no hw-switch switch [switch-number] logging onboard [message level]** global configuration command.
- To clear all the OBFL data in the flash memory except for the uptime and CLI command information, use the **clear onboard switch switch-number** privileged EXEC command.
- In a switch stack, you can enable OBFL on a standalone switch or on all stack members by using the **hw-switch switch [switch-number] logging onboard [message level level]** global configuration command.
- You can enable or disable OBFL on a member switch from the active stack.

For more information about the commands in this section, see the command reference for this release.

## Verifying Troubleshooting of the Software Configuration

### Displaying OBFL Information

*Table 187: Commands for Displaying OBFL Information*

| Command                                                                                             | Purpose                                                                                                 |
|-----------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------|
| <b>show logging onboard [module[switch-number ]]clilog</b><br>Device# show logging onboard 1 clilog | Displays the OBFL CLI commands that were entered on a standalone switch or the specified stack members. |

| Command                                                                                                                     | Purpose                                                                                                                                                                                                                                                                             |
|-----------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>show logging onboard [module[switch-number ]] environment</b><br>Device# show logging onboard 1 environment              | Displays the UDI information for a standalone switch or the specified stack members and for all the connected FRU devices: the PID, the VID, and the serial number.                                                                                                                 |
| <b>show logging onboard [module[switch-number ]] message</b><br>Device# show logging onboard 1 message                      | Displays the hardware-related messages generated by a standalone switch or the specified stack members.                                                                                                                                                                             |
| <b>show logging onboard [module[switch-number ]] poe</b><br>Device# show logging onboard 1 poe                              | Displays the power consumption of PoE ports on a standalone switch or the specified stack members.                                                                                                                                                                                  |
| <b>show logging onboard [module[switch-number ]] temperature</b><br>Device# show logging onboard 1 temperature              | Displays the temperature of a standalone switch or the specified stack members.                                                                                                                                                                                                     |
| <b>show logging onboard [module[switch-number ]] uptime</b><br>Device# show logging onboard 1 uptime                        | Displays the time when a standalone switch or the specified stack members start, the reason the standalone switch or specified stack members restart, and the length of time that the standalone switch or the specified stack members have been running since they last restarted. |
| <b>show logging onboard [module[switch-number ]] voltage</b><br>Device# show logging onboard 1 voltage                      | Displays the system voltages of a standalone switch or the specified stack members.                                                                                                                                                                                                 |
| <b>show logging onboard [module[switch-number ]] continuous</b><br>Device# show logging onboard 1 continuous                | Displays the data in the continuous file.                                                                                                                                                                                                                                           |
| <b>show logging onboard [module[switch-number ]] detail</b><br>Device# show logging onboard 1 detail                        | Displays both the continuous and summary data .                                                                                                                                                                                                                                     |
| <b>show logging onboard [module[switch-number ]] endhh:mm:ss</b><br>Device# show logging onboard 1<br>end 13:00:15 jul 2013 | Displays end time and date on a standalone switch or the specified stack members.                                                                                                                                                                                                   |

| Command                                                                                                                     | Purpose                                                                                 |
|-----------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------|
| <b>show logging onboard [module[switch-number ]]</b><br><br>Device# show logging<br>onboard 1                               | Displays OBFL information about the specified switches in the system.                   |
| <b>show logging onboard [module[switch-number ]] raw</b><br><br>Device# show logging<br>onboard 1 raw                       | Displays the raw information on a standalone switch or the specified stack members.     |
| <b>show logging onboard [module[switch-number ]] start</b><br><br>Device# show logging<br>onboard 1 start 13:00:10 jul 2013 | Displays the start time and date on a standalone switch or the specified stack members. |
| <b>show logging onboard [module[switch-number ]] status</b><br><br>Device# show logging onboard 1 status                    | Displays status information on a standalone switch or the specified stack members.      |
| <b>show logging onboard [module[switch-number ]] summary</b><br><br>Device# show logging onboard 1 summary                  | Displays both the data in the summary file                                              |

For more information, see the *Catalyst 2960-X Switch System Management Command Reference*.

## Example: Verifying the Problem and Cause for High CPU Utilization

To determine if high CPU utilization is a problem, enter the **show processes cpu sorted** privileged EXEC command. Note the underlined information in the first line of the output example.

```
Device# show processes cpu sorted
CPU utilization for five seconds: 8%/0%; one minute: 7%; five minutes: 8%
PID Runtime(ms) Invoked uSecs 5Sec 1Min 5Min TTY Process
309 42289103 752750 56180 1.75% 1.20% 1.22% 0 RIP Timers
140 8820183 4942081 1784 0.63% 0.37% 0.30% 0 HRPC qos request
100 3427318 16150534 212 0.47% 0.14% 0.11% 0 HRPC pm-counters
192 3093252 14081112 219 0.31% 0.14% 0.11% 0 Spanning Tree
143 8 37 216 0.15% 0.01% 0.00% 0 Exec
...
<output truncated>
```

This example shows normal CPU utilization. The output shows that utilization for the last 5 seconds is 8%/0%, which has this meaning:

- The total CPU utilization is 8 percent, including both time running Cisco IOS processes and time spent handling interrupts.
- The time spent handling interrupts is zero percent.

**Table 188: Troubleshooting CPU Utilization Problems**

| <b>Type of Problem</b>                                                           | <b>Cause</b>                                                                                                                    | <b>Corrective Action</b>                                                                                                                       |
|----------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------|
| Interrupt percentage value is almost as high as total CPU utilization value.     | The CPU is receiving too many packets from the network.                                                                         | Determine the source of the network packet. Stop the flow, or change the switch configuration. See the section on “Analyzing Network Traffic.” |
| Total CPU utilization is greater than 50% with minimal time spent on interrupts. | One or more Cisco IOS process is consuming too much CPU time. This is usually triggered by an event that activated the process. | Identify the unusual event, and troubleshoot the root cause. See the section on “Debugging Active Processes.”                                  |

# Scenarios for Troubleshooting the Software Configuration

## Scenarios to Troubleshoot Power over Ethernet (PoE)

Table 189: Power over Ethernet Troubleshooting Scenarios

| Symptom or Problem                                                                                                                                          | Possible Cause and Solution                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |
|-------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <p>Only one port does not have PoE.</p> <p>Trouble is on only one switch port. PoE and non-PoE devices do not work on this port, but do on other ports.</p> | <p>Verify that the powered device works on another PoE port.</p> <p>Use the <b>show run</b>, or <b>show interface status</b> user EXEC commands to verify that the port is not shut down or error-disabled.</p> <p><b>Note</b> Most switches turn off port power when the port is shut down, even though the IEEE specifications make this optional.</p> <p>Verify that <b>power inline never</b> is not configured on that interface or port.</p> <p>Verify that the Ethernet cable from the powered device to the switch port is good: Connect a known good non-PoE Ethernet device to the Ethernet cable, and make sure that the powered device establishes a link and exchanges traffic with another host.</p> <p><b>Note</b> Cisco powered device works only with straight cable and not with crossover one.</p> <p>Verify that the total cable length from the switch front panel to the powered device is not more than 100 meters.</p> <p>Disconnect the Ethernet cable from the switch port. Use a short Ethernet cable to connect a known good Ethernet device directly to this port on the switch front panel (not on a patch panel). Verify that it can establish an Ethernet link and exchange traffic with another host, or ping the port VLAN SVI. Next, connect a powered device to this port, and verify that it powers on.</p> <p>If a powered device does not power on when connected with a patch cord to the switch port, compare the total number of connected powered devices to the switch power budget (available PoE). Use the <b>show power inline</b> command to verify the amount of available power.</p> |

| Symptom or Problem                                                                                                                                                                                        | Possible Cause and Solution                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |
|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <p>No PoE on all ports or a group of ports.</p> <p>Trouble is on all switch ports.</p> <p>Nonpowered Ethernet devices cannot establish an Ethernet link on any port, and PoE devices do not power on.</p> | <p>If there is a continuous, intermittent, or reoccurring alarm related to power, replace the power supply if possible it is a field-replaceable unit. Otherwise, replace the switch.</p> <p>If the problem is on a consecutive group of ports but not all ports, the power supply is probably not defective, and the problem could be related to PoE regulators in the switch.</p> <p>Use the <b>show log</b> privileged EXEC command to review alarms or system messages that previously reported PoE conditions or status changes.</p> <p>If there are no alarms, use the <b>show interface status</b> command to verify that the ports are not shut down or error-disabled. If ports are error-disabled, use the <b>shut</b> and <b>no shut</b> interface configuration commands to reenables the ports.</p> <p>Use the <b>show env power</b> and <b>show power inline</b> privileged EXEC commands to review the PoE status and power budget (available PoE).</p> <p>Review the running configuration to verify that <b>power inline never</b> is not configured on the ports.</p> <p>Connect a nonpowered Ethernet device directly to a switch port. Use only a short patch cord. Do not use the existing distribution cables. Enter the <b>shut</b> and <b>no shut</b> interface configuration commands, and verify that an Ethernet link is established. If this connection is good, use a short patch cord to connect a powered device to this port and verify that it powers on. If the device powers on, verify that all intermediate patch panels are correctly connected.</p> <p>Disconnect all but one of the Ethernet cables from switch ports. Using a short patch cord, connect a powered device to only one PoE port. Verify the powered device does not require more power than can be delivered by the switch port.</p> <p>Use the <b>show power inline</b> privileged EXEC command to verify that the powered device can receive power when the port is not shut down. Alternatively, watch the powered device to verify that it powers on.</p> <p>If a powered device can power on when only one powered device is connected to the switch, enter the <b>shut</b> and <b>no shut</b> interface configuration commands on the remaining ports, and then reconnect the Ethernet cables one at a time to the switch PoE ports. Use the <b>show interface status</b> and <b>show power inline</b> privileged EXEC commands to monitor inline power statistics and port status.</p> <p>If there is still no PoE at any port, a fuse might be open in the PoE section of the power supply. This normally produces an alarm. Check the log again for alarms reported earlier by system messages.</p> |

| Symptom or Problem                                                                                                                                                                                                                                                          | Possible Cause and Solution                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |
|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <p>Cisco pre-standard powered device disconnects or resets.</p> <p>After working normally, a Cisco phone intermittently reloads or disconnects from PoE.</p>                                                                                                                | <p>Verify all electrical connections from the switch to the powered device. Any unreliable connection results in power interruptions and irregular powered device functioning such as erratic powered device disconnects and reloads.</p> <p>Verify that the cable length is not more than 100 meters from the switch port to the powered device.</p> <p>Notice what changes in the electrical environment at the switch location or what happens at the powered device when the disconnect occurs.</p> <p>Notice whether any error messages appear at the same time a disconnect occurs. Use the <b>show log</b> privileged EXEC command to review error messages.</p> <p>Verify that an IP phone is not losing access to the Call Manager immediately before the reload occurs. (It might be a network problem and not a PoE problem.)</p> <p>Replace the powered device with a non-PoE device, and verify that the device works correctly. If a non-PoE device has link problems or a high error rate, the problem might be an unreliable cable connection between the switch port and the powered device.</p> |
| <p>IEEE 802.3af-compliant or IEEE 802.3at-compliant powered devices do not work on Cisco PoE switch.</p> <p>A non-Cisco powered device is connected to a Cisco PoE switch, but never powers on or powers on and then quickly powers off. Non-PoE devices work normally.</p> | <p>Use the <b>show power inline</b> command to verify that the switch power budget (available PoE) is not depleted before or after the powered device is connected. Verify that sufficient power is available for the powered device type before you connect it.</p> <p>Use the <b>show interface status</b> command to verify that the switch detects the connected powered device.</p> <p>Use the <b>show log</b> command to review system messages that reported an overcurrent condition on the port. Identify the symptom precisely: Does the powered device initially power on, but then disconnect? If so, the problem might be an initial surge-in (or <i>inrush</i>) current that exceeds a current-limit threshold for the port.</p>                                                                                                                                                                                                                                                                                                                                                                    |

## Configuration Examples for Troubleshooting Software

### Example: Pinging an IP Host

This example shows how to ping an IP host:

```
Device# ping 172.20.52.3
```

```
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echoes to 172.20.52.3, timeout is 2 seconds:
!!!!
```

```
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/4 ms
Device#
```

**Table 190: Ping Output Display Characters**

| Character | Description                                                               |
|-----------|---------------------------------------------------------------------------|
| !         | Each exclamation point means receipt of a reply.                          |
| .         | Each period means the network server timed out while waiting for a reply. |
| U         | A destination unreachable error PDU was received.                         |
| C         | A congestion experienced packet was received.                             |
| I         | User interrupted test.                                                    |
| ?         | Unknown packet type.                                                      |
| &         | Packet lifetime exceeded.                                                 |

To end a ping session, enter the escape sequence (**Ctrl-^ X** by default). Simultaneously press and release the **Ctrl**, **Shift**, and **6** keys and then press the **X** key.

## Example: Performing a Traceroute to an IP Host

This example shows how to perform a **traceroute** to an IP host:

```
Device# traceroute ip 192.0.2.10

Type escape sequence to abort.
Tracing the route to 192.0.2.10

 0 192.0.2.1 0 msec 0 msec 4 msec
 1 192.0.2.203 12 msec 8 msec 0 msec
 2 192.0.2.100 4 msec 0 msec 0 msec
 3 192.0.2.10 0 msec 4 msec 0 msec
```

The display shows the hop count, the IP address of the router, and the round-trip time in milliseconds for each of the three probes that are sent.

**Table 191: Traceroute Output Display Characters**

| Character | Description                                                                                       |
|-----------|---------------------------------------------------------------------------------------------------|
| *         | The probe timed out.                                                                              |
| ?         | Unknown packet type.                                                                              |
| A         | Administratively unreachable. Usually, this output means that an access list is blocking traffic. |
| H         | Host unreachable.                                                                                 |
| N         | Network unreachable.                                                                              |



| Character | Description           |
|-----------|-----------------------|
| P         | Protocol unreachable. |
| Q         | Source quench.        |
| U         | Port unreachable.     |

To end a trace in progress, enter the escape sequence (**Ctrl-^ X** by default). Simultaneously press and release the **Ctrl**, **Shift**, and **6** keys and then press the **X** key.

## Example: Enabling All System Diagnostics



**Caution** Because debugging output takes priority over other network traffic, and because the **debug all** privileged EXEC command generates more output than any other **debug** command, it can severely diminish switch performance or even render it unusable. In virtually all cases, it is best to use more specific **debug** commands.

This command disables all-system diagnostics:

```
Device# debug all
```

The **no debug all** privileged EXEC command disables all diagnostic output. Using the **no debug all** command is a convenient way to ensure that you have not accidentally left any **debug** commands enabled.

## Additional References for Troubleshooting Software Configuration

### Related Documents

| Related Topic                                  | Document Title                                                                      |
|------------------------------------------------|-------------------------------------------------------------------------------------|
| Troubleshooting commands                       | <i>Catalyst 2960-XR Switch System Management Command Reference</i>                  |
| Interface and hardware component configuration | <i>Catalyst 2960-XR Switch Interface and Hardware Component Configuration Guide</i> |
| Platform-independent command references        | <i>Cisco IOS 15.3M&amp;T Command References</i>                                     |
| Platform-independent configuration information | <i>Cisco IOS 15.3M&amp;T Configuration Guides</i>                                   |

**Standards and RFCs**

| Standard/RFC | Title |
|--------------|-------|
| None         | —     |

**MIBs**

| MIB                                  | MIBs Link                                                                                                                                                                                                              |
|--------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| All supported MIBs for this release. | To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL:<br><a href="http://www.cisco.com/go/mibs">http://www.cisco.com/go/mibs</a> |

**Technical Assistance**

| Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             | Link                                                                    |
|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------|
| <p>The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies.</p> <p>To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds.</p> <p>Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.</p> | <a href="http://www.cisco.com/support">http://www.cisco.com/support</a> |

## Feature History and Information for Troubleshooting Software Configuration

| Release                      | Modification                 |
|------------------------------|------------------------------|
| Cisco IOS Release 15.0(2)EX1 | This feature was introduced. |



## CHAPTER 98

# Information About Licensing

---

- [Restrictions for Configuring Licenses, on page 1969](#)
- [Information About Licensing, on page 1969](#)
- [How to Configure Add-On License Levels, on page 1972](#)
- [Configuration Examples for License Levels, on page 1974](#)
- [Feature History for Information About Licensing, on page 1976](#)

## Restrictions for Configuring Licenses

- Members of a switch stack must run the same license level (base license level and add-on). If the license level is different with a mismatched base license, the switch will not join the stack until it is changed and rebooted from the active stack. Mismatched add-on licenses are automatically synced by the active stack.
- A permanent license can be moved from one device to another. To activate a license, you must reboot your switch.
- An expired evaluation license cannot be reactivated after reboot.

## Information About Licensing

### Overview of License Levels

Software features on the switch are available with base (also known as feature sets) and add-on license levels. Their validity duration determines the license type.

- **Base license levels** for a switch are indicated by the switch model number. They are always permanent licenses, without an expiration date.
- **Add-on license levels** provide Cisco innovations on the switch, as well as on the Cisco Digital Network Architecture Center (Cisco DNA Center). Add-on licenses may be ordered only with a term license type, for a three, five, or seven year period.

## Base Licenses

The switch is shipped with an IP Lite base license. The model number is an indicator of the license level. See the last suffix in the model number. -I indicates an IP Lite model. For example, WS-C2960XR-48FPS-I has an IP Lite Image.




---

**Note** The base license level is tied to the hardware model and cannot be changed.

---

## Add-On Licenses

The DNA essentials add-on license is available.

The following guidelines apply to Add-on Licenses:

- A Reboot is not required when you configure an add-on license.
- Add-on licenses may be ordered for a three, five, or seven year period.
- You must set up Cisco SSM to receive daily e-mail alerts, to be notified of expiring add-on licenses that you want to renew.
- Only the DNA Essentials add-on license is available. (Although visible on the CLI, the DNA Advantage license level is not available).

## License States

You can also access the license information by using the **show license** command in the privileged EXEC mode.

*Table 192: Right-to-use license states*

| License State      | Description                                                                   |
|--------------------|-------------------------------------------------------------------------------|
| Active, In Use     | EULA was accepted and the license is in use after device reboot.              |
| Active, Not In Use | EULA was accepted and the switch is ready to use when the license is enabled. |
| Not Activated      | EULA was not accepted.                                                        |

The following example shows how to display the license level of the switch. The example shows LAN Base as the active license and as the one that is in use.

```
Switch# show license

Index 1
License Name : lanlite
Period left : 0 minute 0 second
License Type: Permanent
License State: Inactive
Index 2
```

```
License Name : lanbase
Period left : 0 minute 0 second
License Type : Permanent
License State : Active, In use
```

```
Index 3
License Name : dna-essentials
Period left : CSSM Managed
License Type : Subscription
License State : Active, In use
```

```
Index 4
License Name : dna-advantage
Period left : CSSM Managed
License Type : Subscription
License State : Not Activated
```

Guidelines to follow when monitoring your image based license state:

- A purchased permanent license is set to Active, In Use state only after a switch reboot.
- If more than one license was purchased, a reboot will activate the license with the highest feature set. For instance, the LAN Base license is activated and not the LAN Lite license.
- The remaining licenses purchased after switch reboot, stay in Active, Not In Use state.

## Guidelines for License Types

Licenses may be of the permanent or term type only.

- Permanent: For a license level, and without an expiration date. The basic license type for the switch is determined by the model and is always permanent.
- Term: For a license level, and for a three, five, or seven year period. Add-on licenses (DNA Essentials and DNA Advantage) may be ordered only with a term license type.

## Ordering with Smart Accounts

We recommend that you use Smart Accounts to order devices as well as licenses. Smart Accounts enable you to manage all of your software licenses for switches, routers, firewalls, access-points or tools from one centralized website. To create Smart Accounts, use the Cisco Smart Software Manager (Cisco SSM).



---

**Note** This is especially relevant to the term licenses that you order, because information about the expiry of term licenses is available only through the Cisco SSM website.

---

For more information about Cisco SSM, see: <http://www.cisco.com/c/en/us/buy/smart-accounts/software-licensing.html>

## License Activation for Switch Stacks

LAN Base models can stack with LAN Base models only.

The active stack is activated with a license from its active console. The license level for members in the stack can be activated at the same time.

To change the license level, do not disconnect a newly added stack member if the stack cables are connected. Instead, use the active console to set the new member's license level at the same license level as an active stack and reboot the new member to join the stack.

Reboot is required only for the base license; not when you configure an add-on license

## How to Configure Add-On License Levels

The following sections provide information on how to configure Add-on License Levels.

### Activating an Image Based Add-on License

The following steps can be used to activate an image based license.

#### SUMMARY STEPS

1. `enable`
2. `configure terminal`
3. `license boot level addon addon-license`
4. `license accept end user agreement force`
5. `show license right-to-use usage`

#### DETAILED STEPS

|        | Command or Action                                                                                                                               | Purpose                                                                                                                                     |
|--------|-------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------|
| Step 1 | <b>enable</b><br><b>Example:</b><br>Device> <code>enable</code>                                                                                 | Enables privileged EXEC mode. Enter your password if prompted.                                                                              |
| Step 2 | <b>configure terminal</b><br><b>Example:</b><br>Device# <code>configure terminal</code>                                                         | Enters global configuration mode.                                                                                                           |
| Step 3 | <b>license boot level addon <i>addon-license</i></b><br><b>Example:</b><br>Device(config)# <code>license boot level addon dna-essentials</code> | Specifies the add-on license level. The following options are available: <ul style="list-style-type: none"> <li>• DNA Essentials</li> </ul> |
| Step 4 | <b>license accept end user agreement force</b><br><b>Example:</b>                                                                               | Enables acceptance of the end-user license agreement (EULA).                                                                                |

|               | Command or Action                                                                                                | Purpose                                                                                                                                                                   |
|---------------|------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|               | Device(config)# license accept end user agreement<br>force                                                       | <b>Note</b> To configure an add-on license EULA acceptance is not mandatory, but you will not be able to use or configure the DNAC features until you complete this step. |
| <b>Step 5</b> | <b>show license right-to-use usage</b><br><br><b>Example:</b><br>Device(config)# show license right-to-use usage | Displays detailed usage information.<br><br>Other options are available with the <b>show license right-to-use command</b> .                                               |

## Rehosting a License

To rehost a license, you have to deactivate the license from one device and then activate the same license on another device. The following steps can be used to rehost a license.

### SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **license right-to-use deactivate [license-level] slotslot-num**
4. **license right-to-use activate [license-level]slot-num [acceptEULA]**

### DETAILED STEPS

|               | Command or Action                                                                                                                                                          | Purpose                                                                 |
|---------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------|
| <b>Step 1</b> | <b>enable</b><br><br><b>Example:</b><br>Device> enable                                                                                                                     | Enables privileged EXEC mode.<br><br>• Enter your password if prompted. |
| <b>Step 2</b> | <b>configure terminal</b><br><br><b>Example:</b><br>Device# configure terminal                                                                                             | Enters global configuration mode.                                       |
| <b>Step 3</b> | <b>license right-to-use deactivate [license-level] slotslot-num</b><br><br><b>Example:</b><br>Device(config)# license right-to-use deactivate<br>dna-essentials slot 1     | Deactivates the license on one device.                                  |
| <b>Step 4</b> | <b>license right-to-use activate [license-level]slot-num [acceptEULA]</b><br><br><b>Example:</b><br>Device(config)# license right-to-use activate<br>dna-essentials slot 2 | Activates the license on another device.                                |

## Monitoring Licenses

Use the following commands in the privilege EXEC mode to monitor license information:

| Command                                                     | Purpose                                                                         |
|-------------------------------------------------------------|---------------------------------------------------------------------------------|
| <b>show license right-to-use default</b>                    | Displays the default license information.                                       |
| <b>show license right-to-use detail</b>                     | Displays detailed information of all the licenses in the switch stack.          |
| <b>show license right-to-use eula</b>                       | Displays the end user license agreement.                                        |
| <b>show license right-to-use slot slot-number</b>           | Displays the license information for a specific slot in a switch stack.         |
| <b>show license right-to-use summary</b>                    | Displays a summary of the license information on the entire switch stack.       |
| <b>show license right-to-use usage [ slot slot-number ]</b> | Displays detailed information about usage for all licenses in the switch stack. |

## Configuration Examples for License Levels

The following sections provide examples for configuring license levels.

### Reference

.

### Example: Displaying the detailed license information

The following examples shows how to display the detailed information of all the licenses in a stack using the **show license right-to-use detail** command.

```
Device# show license right-to-use detail
Index 1
 License Name : Advanced Enterprise Services
 Period left : Lifetime
 License Type : permanent
 License State : Active, In use
Index 2
 License Name : dna-essentials
 Period left : CSSM Managed
 License Type : Subscription
 License State : Not Activated
Index 3
 License Name : dna-advantage
 Period left : CSSM Managed
 License Type : Subscription
 License State : Active, In use
```



## Example: Displaying a summary of the license information

The following examples shows how to display a summary of the license information using the **show license right-to-use summary** command.

```
Device# show license right-to-use summary
License Name Type Period left

lanlite Permanent 0 minute 0 second
lanbase Permanent 0 minute 0 second
dna-essentials Subscription CSSM Managed

```

```
License Level In Use: lanbase addon: dna-essentials
License Level on Reboot: lanbase addon: dna-essentials
```

Example: show license right-to-use usage

```
FEX-0#show license right-to-use usage
slot License Name Type In-use EULA

0 lanlite Permanent yes yes
0 lanbase Permanent yes yes
 dna-essentials Subscription yes yes
 dna-advantage Subscription no yes
```

## Example: Displaying the end user license agreement

The following example shows how to display the end user license agreement.

```
Device# show license right-to-use eula subscription
Feature name EULA Accepted

dna-essentials yes
dna-advantage no
```

```
PLEASE READ THE FOLLOWING TERMS CAREFULLY. INSTALLING THE LICENSE OR
LICENSE KEY PROVIDED FOR ANY CISCO SOFTWARE PRODUCT, PRODUCT FEATURE,
AND OR SUBSEQUENTLY PROVIDED SOFTWARE FEATURES (COLLECTIVELY, THE ?SOFTWARE?),
USING SUCH SOFTWARE, AND/OR ACTIVATION OF THE SOFTWARE COMMAND LINE INTERFACE
CONSTITUTES YOUR FULL ACCEPTANCE OF THE FOLLOWING TERMS.YOU MUST NOT PROCEED
FURTHER IF YOU ARE NOT WILLING TO BE BOUND BY ALL THE TERMS SET FORTH HEREIN.
```

```
Your use of the Software is subject to the Cisco End User License Agreement (EULA)
and any relevant supplemental terms (SEULA) found at
http://www.cisco.com/c/en/us/about/legal/cloud-and-software/software-terms.html.
You hereby acknowledge and agree that certain Software and/or features are licensed
for a particular term, that the license to such Software and/or features is valid only
for the applicable term and that such Software and/or features may be shut down or
otherwise terminated by Cisco after expiration of the applicable license term (e.g.,
90-day trial period). Cisco reserves the right to terminate any such Software feature
electronically or by any other means available. While Cisco may provide alerts, it is
your sole responsibility to monitor your usage of any such term Software feature to
ensure that your systems and networks are prepared for a shutdown of the Software feature.
To memorialize your acceptance of these terms and activate your license to use the Software,
please execute the command "license accept end user agreement force".
```

## Feature History for Information About Licensing

| Release                     | Modification                 |
|-----------------------------|------------------------------|
| Cisco IOS Release 15.2(6)E1 | This feature was introduced. |



## PART **XVI**

# Embedded Event Manager

- [Embedded Event Manager Overview, on page 1979](#)
- [Information About Writing EEM Policies Using the Cisco IOS CLI, on page 2001](#)
- [Writing Embedded Event Manager Policies Using Tcl, on page 2075](#)
- [Signed Tcl Scripts, on page 2135](#)
- [EEM CLI Library Command Extensions, on page 2159](#)
- [EEM Context Library Command Extensions, on page 2171](#)
- [EEM Event Registration Tcl Command Extensions, on page 2179](#)
- [EEM Event Tcl Command Extensions, on page 2263](#)
- [EEM Library Debug Command Extensions, on page 2271](#)
- [EEM Multiple Event Support Tcl Command Extensions, on page 2273](#)
- [EEM SMTP Library Command Extensions, on page 2277](#)
- [EEM System Information Tcl Command Extensions, on page 2281](#)
- [EEM Utility Tcl Command Extensions, on page 2293](#)





## CHAPTER 99

# Embedded Event Manager Overview

Embedded Event Manager (EEM) is a distributed and customized approach to event detection and recovery offered directly in a Cisco IOS device. EEM offers the ability to monitor events and take informational, corrective, or any desired EEM action when the monitored events occur or when a threshold is reached. An EEM policy is an entity that defines an event and the actions to be taken when that event occurs.

This module contains a technical overview of EEM. EEM can be used alone, or with other network management technologies to help monitor and maintain your network. Before you begin to implement EEM, it is important that you understand the information presented in this module.

- [Information About Embedded Event Manager, on page 1979](#)
- [Where to Go Next, on page 1997](#)
- [Feature Information for Embedded Event Manager 4.0 Overview, on page 1997](#)
- [Additional References, on page 1997](#)

## Information About Embedded Event Manager

### Embedded Event Manager

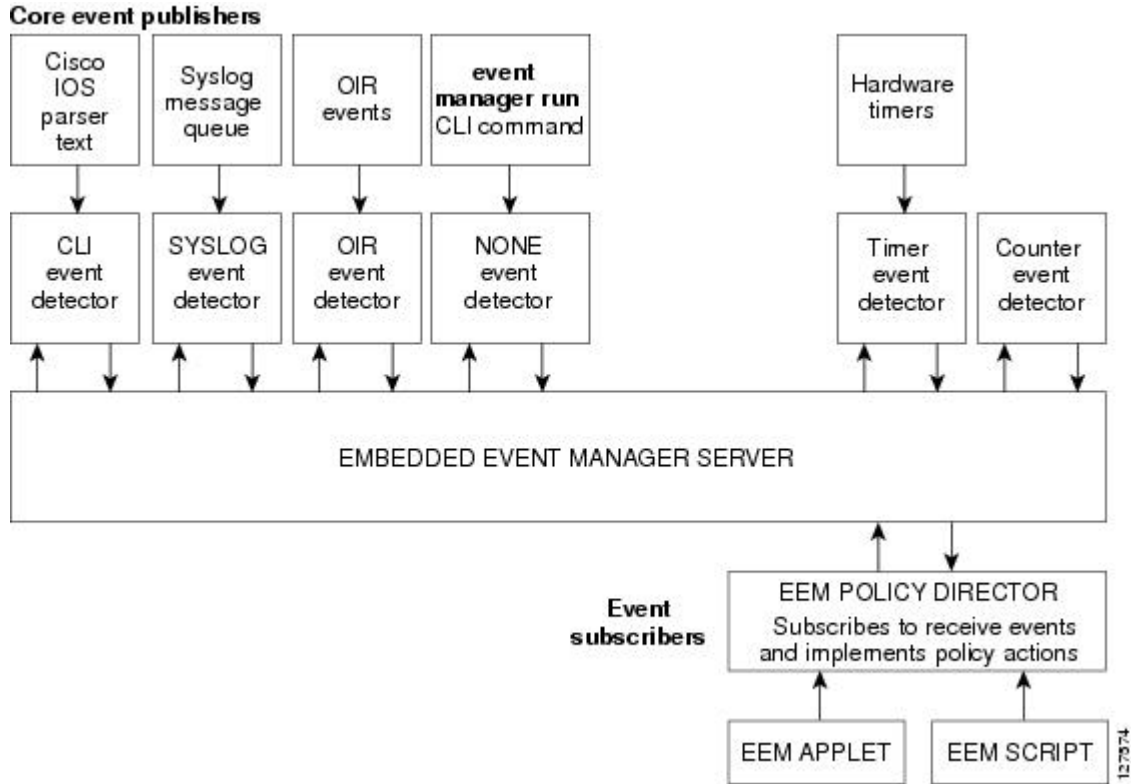
Event tracking and management has traditionally been performed by devices external to the networking device. Embedded Event Manager (EEM) has been designed to offer event management capability directly in Cisco IOS devices. The on-device, proactive event management capabilities of EEM are useful because not all event management can be done off device because some problems compromise communication between the device and the external network management device. Capturing the state of the device during such situations can be invaluable in taking immediate recovery actions and gathering information to perform root-cause analysis. Network availability is also improved if automatic recovery actions are performed without the need to fully reboot the routing device.

EEM is a flexible, policy-driven framework that supports in-box monitoring of different components of the system with the help of software agents known as event detectors. The figure below shows the relationship between the EEM server, core event publishers (event detectors), and the event subscribers (policies). Basically, event publishers screen events and publish them when there is a match on an event specification that is provided by the event subscriber. Event detectors notify the EEM server when an event of interest occurs. The EEM policies that are configured using the Cisco command-line interface (CLI) then implement recovery on the basis of the current state of the system and the actions specified in the policy for the given event.

EEM offers the ability to monitor events and take informational or corrective action when the monitored events occur or when a threshold is reached. An EEM policy is an entity that defines an event and the actions

to be taken when that event occurs. There are two types of EEM policies: an applet or a script. An applet is a simple form of policy that is defined within the CLI configuration. A script is a form of policy that is written in Tool Command Language (Tcl).

Figure 140: Embedded Event Manager Core Event Detectors



**Note** If your network has a higher version of EEM, that version would include the previous releases of EEM version as well.

## Embedded Event Manager 1.0

EEM 1.0 introduced the following event detectors:

- **SNMP**—The Simple Network Management Protocol (SNMP) event detector allows a standard SNMP MIB object to be monitored and an event to be generated when the object matches specified values or crosses specified thresholds.
- **Syslog**—The syslog event detector allows for screening syslog messages for a regular expression pattern match.

EEM 1.0 introduced the following actions:

- Generating prioritized syslog messages.
- Generating a Cisco Networking Services (CNS) event for upstream processing by CNS devices.

- Reloading the Cisco software.
- Switching to a secondary processor in a fully redundant hardware configuration.

## Embedded Event Manager 2.0

EEM 2.0 introduced some new features. EEM 2.0 introduced the following event detectors:

- **Application-Specific**—The application-specific event detector allows any Embedded Event Manager policy to publish an event.
- **Counter**—The counter event detector publishes an event when a named counter crosses a specified threshold.
- **Interface Counter**—The interface counter event detector publishes an event when a generic Cisco IOS interface counter for a specified interface crosses a defined threshold.
- **Timer**—The timer event detector publishes events for the following four different types of timers: absolute-time-of-day, countdown, watchdog, and CRON.
- **Watchdog System Monitor (IOSWDSysMon)**—The Cisco IOS watchdog system monitor event detector publishes an event when CPU or memory utilization for a Cisco IOS process crosses a threshold.

EEM 2.0 introduced the following actions:

- Setting or modifying a named counter.
- Publishing an application-specific event
- Generating an SNMP trap.

The ability to run a Cisco defined sample policy written using Tool Command Language (Tcl) was introduced. A sample policy was provided that could be stored in the system policy directory.

## Embedded Event Manager 2.1

EEM 2.1 introduced the following new event detectors:

- **CLI**—The CLI event detector screens command-line interface (CLI) commands for a regular expression match.
- **None**—The none event detector publishes an event when the Cisco IOS **event manager run** command executes an EEM policy.
- **OIR**—The online insertion and removal (OIR) event detector publishes an event when a particular hardware insertion or removal event occurs.

EEM 2.1 introduced the following actions:

- Executing a Cisco CLI command.
- Requesting system information when an event occurs.
- Sending a short e-mail.
- Manually running an EEM policy.

EEM 2.1 also permits multiple concurrent policies to be run using the new **event manager scheduler script** command. Support for SNMP event detector rate-based events is provided as is the ability to create policies using Tool Command Language (Tcl).

## Embedded Event Manager 2.1 (Software Modularity)

EEM 2.1 (Software Modularity) is supported on Cisco Software Modularity images. EEM 2.1 (Software Modularity) introduced the following event detectors:

- **GOLD**—The Generic Online Diagnostic (GOLD) event detector publishes an event when a GOLD failure event is detected on a specified card and subcard.
- **System Manager**—The system manager event detector generates events for Cisco IOS Software Modularity process start, normal or abnormal stop, and restart events. The events generated by the system manager allows policies to change the default behavior of the process restart.
- **Watchdog System Monitor (WDSysMon)**—The Cisco Software Modularity watchdog system monitor event detector detects infinite loops, deadlocks, and memory leaks in Cisco IOS Software Modularity processes.

EEM 2.1 for Software Modularity introduced the ability to display EEM reliability metric data for processes.



---

**Note** EEM 2.1 for Software Modularity images also supports the resource and RF event detectors introduced in EEM 2.2, but it does not support the enhanced object tracking event detector or the actions to read and set tracked objects.

---

## Embedded Event Manager 2.2

EEM 2.2 introduced some new features. EEM 2.2 introduced the following event detectors:

- **Enhanced Object Tracking**—The enhanced object tracking event detector publishes an event when the tracked object changes. Enhanced object tracking provides complete separation between the objects to be tracked and the action to be taken by a client when a tracked object changes.
- **Resource**—The resource event detector publishes an event when the Embedded Resource Manager (ERM) reports an event for the specified policy.
- **RF**—The redundancy framework (RF) event detector publishes an event when one or more RF events occur during synchronization in a dual Route Processor (RP) system. The RF event detector can also detect an event when a dual RP system continuously switches from one RP to another RP (referred to as a ping-pong situation).

EEM 2.2 introduced the following actions:

- Reading the state of a tracked object.
- Setting the state of a tracked object.



## Embedded Event Manager 2.3

EEM 2.3 is supported on the Cisco Catalyst 6500 Series switches and introduces enhancements to the Generic Online Diagnostics (GOLD) Event Detector on that product.

- The **event gold** command was enhanced with the addition of the **action-notify**, **testing-type**, **test-name**, **test-id**, **consecutive-failure**, **platform-action**, and **maxrun** keywords for improved reaction to GOLD test failures and conditions.
- The following platform-wide GOLD Event Detector information can be accessed through new read-only EEM built-in environment variables:
  - Boot-up diagnostic level
  - Card index, name, serial number
  - Port counts
  - Test counts
- The following test-specific GOLD Event Detector information can be accessed through new read-only EEM built-in environment variables (available to EEM applets only):
  - Test name, attribute, total run count
  - Test result per test, port, or device
  - Total failure count, last fail time
  - Error code
  - Occurrence of consecutive failures

These enhancements result in reduced mean time to recovery (MTTR) and higher availability through improved automation and fault detection.

## Embedded Event Manager 2.4

EEM 2.4 introduced the following event detectors:

- **SNMP Notification**—The SNMP notification event detector provides the ability to intercept SNMP trap and inform messages coming into the device. An SNMP notification event is generated when an incoming SNMP trap or inform message matches specified values or crosses specified thresholds.
- **RPC**—The remote procedure call (RPC) event detector provides the ability to invoke EEM policies from outside the device over an encrypted connection using Secure Shell (SSH). The RPC event detector uses Simple Object Access Protocol (SOAP) data encoding for exchanging XML-based messages. This event detector can be used to run EEM policies and then receive output in a SOAP XML-formatted reply.

EEM 2.4 added enhancements to the following event detectors:

- **Interface counter rate-based trigger**—This feature adds the ability for an interface event to be triggered based on a rate of change over a period of time. A rate can be specified both for the entry value and the exit value. This feature copies the rate-based functionality that currently exists for the SNMP event detector.
- **SNMP delta value**—The difference between the monitored Object Identifier (OID) value at the beginning of the monitored period and the actual OID value when the event is published will be provided in the **event reqinfo** data for both the SNMP event detector and the Interface Counter event detector.

EEM 2.4 introduced the following actions:

- Multiple event support—The ability to run multiple events was introduced, and **show event manager** commands were enhanced to show multiple events.
- Support for parameters—The *parameter* argument has been added to the **event manager run** command. A maximum of 15 parameters can be used.
- Display of Job IDs and completion status--Some of the **show event manager** commands were enhanced to display Job IDs and completion status.
- Bytecode support—Tcl 8 defines a specialized bytecode language (BCL) and includes a just-in-time compiler that translates Tcl scripts to BCL. Byte sequence is executed by a “virtual machine,” Tcl\_ExecuteByteCode(), or TEBC for short, as often as needed. Currently EEM accepts file extensions, such as \*.tcl for user policies and \*.tm for system policies. Tcl standard extension for bytecode scripts are \*.tbc. Now EEM will accept \*.tbc as valid EEM policies.
- Registration substitution enhancement—Supports replacing multiple parameters in the event registration statement lines with a single environment variable.
- Tcl package support

## Embedded Event Manager 3.0

EEM 3.0 introduced the following new event detectors:

- Custom CLI--The custom CLI event detector publishes an event to add and enhance existing CLI command syntax.
- Routing--The Routing event detector publishes an event when route entries change in the Routing Information Base (RIB).
- NetFlow-- The NetFlow event detector publishes an event when a NetFlow event is triggered.
- IP SLA--The IP SLA event detector publishes an event when an IP SLA reaction is triggered.

EEM 3.0 introduced the following features:

- Class-based scheduling--The EEM policies will be assigned a class using the **class** keyword when they are registered. EEM policies registered without a class will be assigned to the default class.
- High performance Tcl policies--Three new Tcl commands are introduced **event\_completion**, **event\_wait**, and **event\_completion\_with\_wait**.
- Interactive cli support--The synchronous applets are enhanced to support interaction with the local console (TTY). Two new IOS commands, **action gets** and **action puts**, are introduced to allow users to enter and display input directly on the console.
- Variable logic for applets--The Variable Logic for EEM Applets feature adds the ability to apply conditional logic within EEM applets. Conditional logic introduces a control structure that can change the flow of actions within applets depending on conditional expressions.
- Digital signature support--A new API performs digital signature verification for a Tcl script to check it the script is signed by Cisco before execution.
- Support authenticating e-mail servers--The **action mail** command is modified to include an optional username and password.

- SMTP IPv6 support--The keyword **sourceaddr** is added in Tcl e-mail templates to specify either an IPv6 or IPv4 address.
- SNMP library extensions--The EEM applet **action info** and Tcl **sys\_reqinfo\_snmp** commands are enhanced to include functionality for SNMP getid, inform, trap, and set-type operations.
- SNMP Notification IPv6 support--IPv6 address is supported for the source and destination IP addresses.
- CLI Library XML-PI support--Provides a programmable interface which encapsulates IOS command-line interface (CLI) show commands in XML format in a consistent way across different Cisco products. Customers using XML-PI will be able to parse IOS show command output from within Tcl scripts using well-known keywords instead of having to depend on the use of regular expression support.

## Embedded Event Manager 3.1

EEM 3.1 introduced one new event detector:

- SNMP Object--The Simple Network Management Protocol (SNMP) object trap event detector provides an extension to replace the value when an SNMP trap with the specified SNMP object ID (OID) is encountered on a specific interface or address.

EEM 3.1 added an enhancement to the following event detector:

- SNMP Notification--The SNMP notification event detector now can wait and intercept the outgoing SNMP traps and informs.

EEM 3.1 added enhancement to the following action:

- Specify facility--The **action syslog** command has been enhanced to specify syslog facility.

EEM 3.1 introduces the following features:

- Provides the ability to create a short description for the registered policy--A new **description** command has been introduced to register policies with a brief description in Cisco IOS CLI and Tcl policies. The **show event manager policy available** command and the **show event manager policy registered** command have been enhanced to add the **description** keyword to display the description of the registered applet.
- Enables EEM policies to bypass AAA authorization--The **event manager application** command has been enhanced to provide authorization and bypass keywords to disable AAA.
- Introduces CLI Library enhancements--Provides two new commands in the CLI library: **cli\_run** and **cli\_run\_interactive**.

## Embedded Event Manager 3.2

EEM 3.2 introduced the following new event detectors:

- Neighbor Discovery--Neighbor Discovery event detector provides the ability to publish a policy to respond to automatic neighbor detection when:
  - a Cisco Discovery Protocol (CDP) cache entry is added, deleted or updated.
  - a Link Layer Discovery Protocol (LLDP) cache entry is added, deleted, or updated.
  - an interface link status changes.

- an interface line status changes.
- Identity--Identity event detector generates an event when AAA authorization and authentication is successful, when failure occurs, or after normal user traffic on the port is allowed to flow.
- Mac-Address-Table--Mac-Address-Table event detector generates an event when a MAC address is learned in the MAC address table.



**Note** The Mac-Address-Table event detector is supported only on switch platforms and can be used only on Layer 2 interfaces where MAC addresses are learned. Layer 3 interfaces do not learn addresses and devices do not usually support the mac-address-table infrastructure needed to notify EEM of a learned MAC address.

EEM 3.2 also introduces new CLI commands to support the applets to work with the new event detectors.

## Embedded Event Manager 4.0

EEM 4.0 introduces the following new features:

- EEM Email Action Enhancements
  - TLS support for SMTP mail actions—The new optional **secure** keyword is added to the **action mail** CLI with **tls** and **none** keyword options. There are no updates to the corresponding Tcl Policy.
  - Custom port for SMTP mail actions—The new optional **port** keyword is added to the **action mail** CLI. In the Tcl policy, the port number can be specified by adding a line to the e-mail template.
- EEM Security Enhancements
  - Checksum-based script integrity—Where digital signature is not supported or unavailable, users can still enforce some basic integrity check on the TCL policy by using the Unix command **openssl sha1**. The new optional **checksum**, **md5**, and **sha-1** keywords have been added to the **event manager policy** command.
  - Third-party digital signature support—Requires Tcl secure mode and a trustpoint to associate with the TCL scripts in order to verify the signature.
  - Script owner identification—If a policy is successfully registered with a digital signature, the owner (or signer) of the policy can be identified by using the **show event manager policy registered** command and checking the **Dsig** keyword in the show output.
  - Registration of remote Tcl policies—The new optional **remote** keyword has been added to the **event manager policy** command.
- EEM Resource Management
  - Resource consumption throttling—The new optional **resource-limit** keyword has been added to the **event manager scheduler** command.
  - Rate limiting of triggered policies per event—The new optional **rate-limit** keyword has been added to the **event syslog** command.
- EEM Usability Enhancements
  - File operations in EEM applet actions—The new CLI **action file** has been added to allow file selection.
  - New fields are added in EEM to track statistics of queue size, dropped events, and run-time using the **show event manager statistics EXEC** command. A set of new clear commands—**clear event**

**manager detector counters** and **clear event manager server counters** —are introduced to clear the event manager queue counters.

- EEM Event Detector Enhancements
  - CLI event detector enhancement—Provides the ability to detect the session where the user enters the event cli command. Four new keywords and built-in environmental variables—**username**, **host**, **privilege**, and **tty**— are added to the **event cli** applet and event\_reqinfo array names to the **event\_register\_cli** event detector. The **show event manager detector EXEC** command has also been modified to reflect the enhancement.
  - Syslog event detector performance enhancement—Provides the option to perform string matching on specific log message fields. The four new keywords—**facility**, **mnemonic**, **sequence**, and **timestamp** keywords— are added to the **action syslog** command, **event syslog** command, and to the **event\_register\_syslog** event detector. The **show event manager detector EXEC** command has also been modified to reflect the enhancement.

## EEM Event Detectors Available by Cisco IOS Release

EEM uses software programs known as event detectors to determine when an EEM event occurs. Some event detectors are available on every Cisco IOS release, but most event detectors have been introduced in a specific release. Use the table below to determine which event detectors are available in your specific Cisco IOS release. A blank entry (--) indicates that the event detector is not available; the text “Yes” indicates that the event detector is available. The event detectors shown in the table are supported in later releases of the same Cisco IOS release train. For more details on each event detector, see the Event Detectors concept in the “Embedded Event Manager Overview” module.

**Table 193: Availability of Event Detectors by Cisco IOS Release**

| Event Detector           | 12.2(25)S | 12.3(14)T<br>12.2(18)SXF5<br>12.2(28)SB<br>12.2(33)SRA | 12.4(2)T<br>12.2(31)SB3<br>12.2(33)SRB | 12.2(18)SXF4<br>Cisco IOS Software Modularity | 12.2(33)SXH | 12.4(20)T<br>12.2(33)SXI | 12.4(22)T<br>12.2(33)SRE | 15.0(1)M<br>15.1(3)T | 15.2(3)S<br>15.2(3)SY | 15 E<br>XE 3E |
|--------------------------|-----------|--------------------------------------------------------|----------------------------------------|-----------------------------------------------|-------------|--------------------------|--------------------------|----------------------|-----------------------|---------------|
| Application-Specific     | Yes       | Yes                                                    | Yes                                    | Yes                                           | Yes         | Yes                      | Yes                      | Yes                  | Yes                   | Yes           |
| CLI                      | --        | Yes                                                    | Yes                                    | Yes                                           | Yes         | Yes                      | Yes                      | Yes                  | --                    | Yes           |
| Counter                  | Yes       | Yes                                                    | Yes                                    | Yes                                           | Yes         | Yes                      | Yes                      | Yes                  | Yes                   | Yes           |
| Custom CLI               | --        | --                                                     | --                                     | --                                            | --          | --                       | Yes                      | Yes                  | --                    | --            |
| Enhanced Object Tracking | --        | --                                                     | Yes                                    | --                                            | Yes         | Yes                      | Yes                      | Yes                  | --                    | --            |
| Environmental            | --        | --                                                     | --                                     | --                                            | --          | --                       | --                       | --                   | --                    | Yes           |
| GOLD                     | --        | --                                                     | --                                     | Yes                                           | Yes         | Yes                      | Yes                      | Yes                  | --                    | Yes           |
| Identity                 | --        | --                                                     | --                                     | --                                            | --          | --                       | --                       | Yes                  | Yes                   | Yes           |
| Interface Counter        | Yes       | Yes                                                    | Yes                                    | Yes                                           | Yes         | Yes                      | Yes                      | Yes                  | --                    | Yes           |

| Event Detector                                             | 12.2(25)S | 12.3(14)T<br>12.2(18)SXF5<br>12.2(28)SB<br>12.2(33)SRA | 12.4(2)T<br>12.2(31)SB3<br>12.2(33)SRB | 12.2(18)SXF4<br>Cisco IOS<br>Software<br>Modularity | 12.2(33)SXH | 12.4(20)T<br>12.2(33)SXI | 12.4(22)T<br>12.2(33)SRE | 15.0(1)M<br>15.1(3)T | 15.2(5)S<br>15.2(5)SY | 15 E<br>XE 3E |
|------------------------------------------------------------|-----------|--------------------------------------------------------|----------------------------------------|-----------------------------------------------------|-------------|--------------------------|--------------------------|----------------------|-----------------------|---------------|
| IPSLA                                                      | --        | --                                                     | --                                     | --                                                  | --          | --                       | Yes                      | Yes                  | --                    | Yes           |
| Mac-Address-Table                                          | --        | --                                                     | --                                     | --                                                  | --          | --                       | --                       | Yes                  | Yes                   | Yes           |
| Neighbor<br>Discovery                                      | --        | --                                                     | --                                     | --                                                  | --          | --                       | --                       | Yes                  | Yes                   | Yes           |
| NF                                                         | --        | --                                                     | --                                     | --                                                  | --          | --                       | Yes                      | Yes                  | --                    | --            |
| None                                                       | --        | Yes                                                    | Yes                                    | Yes                                                 | Yes         | Yes                      | Yes                      | Yes                  | Yes                   | Yes           |
| OIR                                                        | --        | Yes                                                    | Yes                                    | Yes                                                 | Yes         | Yes                      | Yes                      | Yes                  | Yes                   | Yes           |
| Resource                                                   | --        | --                                                     | Yes                                    | Yes                                                 | Yes         | Yes                      | Yes                      | Yes                  | --                    | --            |
| RF                                                         | --        | --                                                     | Yes                                    | Yes                                                 | Yes         | Yes                      | Yes                      | Yes                  | --                    | Yes           |
| Routing                                                    | --        | --                                                     | --                                     | --                                                  | --          | --                       | Yes                      | Yes                  | --                    | Yes           |
| RPC                                                        | --        | --                                                     | --                                     | --                                                  | --          | Yes                      | Yes                      | Yes                  | Yes                   | --            |
| SNMP                                                       | Yes       | Yes                                                    | Yes                                    | Yes                                                 | Yes         | Yes                      | Yes                      | Yes                  | --                    | Yes           |
| SNMP Proxy                                                 | --        | --                                                     | --                                     | --                                                  | --          | --                       | --                       | --                   | Yes                   | --            |
| SNMP Notification                                          | --        | --                                                     | --                                     | --                                                  | --          | Yes                      | Yes                      | Yes                  | --                    | Yes           |
| SNMP Object                                                | --        | --                                                     | --                                     | --                                                  | --          | --                       | --                       | Yes                  | --                    | Yes           |
| Syslog                                                     | Yes       | Yes                                                    | Yes                                    | Yes                                                 | Yes         | Yes                      | Yes                      | Yes                  | Yes                   | Yes           |
| System Manager                                             | --        | --                                                     | --                                     | Yes                                                 | Yes         | Yes                      | Yes                      | Yes                  | Yes                   | --            |
| Timer                                                      | Yes       | Yes                                                    | Yes                                    | Yes                                                 | Yes         | Yes                      | Yes                      | Yes                  | Yes                   | Yes           |
| IOSWDSysMon<br>(Cisco IOS<br>watchdog)                     | Yes       | Yes                                                    | Yes                                    | Yes                                                 | Yes         | Yes                      | Yes                      | Yes                  | --                    | Yes           |
| WDSysMon (Cisco<br>IOS Software<br>Modularity<br>watchdog) | --        | --                                                     | --                                     | Yes                                                 | --          | --                       | --                       | --                   | --                    | --            |

## Event Detectors

Embedded Event Manager (EEM) uses software programs known as *event detectors* to determine when an EEM event occurs. Event detectors are separate systems that provide an interface between the agent being monitored, for example Simple Network Management Protocol (SNMP), and the EEM policies where an action can be implemented. Some event detectors are available on every Cisco IOS release, but most event detectors have been introduced in a specific release. For details of which event detector is supported in each Cisco IOS release, see the EEM Event Detectors Available by Cisco IOS Release concept in the “Writing Embedded Event Manager Policies Using the Cisco IOS CLI” or the “Writing Embedded Event Manager Policies Using Tcl” modules. EEM contains the following event detectors.

### Application-Specific Event Detector

The application-specific event detector allows any Embedded Event Manager policy to publish an event. When an EEM policy publishes an event it must use an EEM subsystem number of 798 with any event type. If an existing policy is registered for subsystem 798 and a specified event type, a second policy of the same event type will trigger the first policy to run when the specified event is published.

### CLI Event Detector

The CLI event detector screens command-line interface (CLI) commands for a regular expression match. When a match is found, an event is published. The match logic is performed on the fully expanded CLI command after the command is successfully parsed and before it is executed. The CLI event detector supports three publish modes:

- Synchronous publishing of CLI events--The CLI command is not executed until the EEM policy exits, and the EEM policy can control whether the command is executed. The read/write variable, `_exit_status`, allows you to set the exit status at policy exit for policies triggered from synchronous events. If `_exit_status` is 0, the command is skipped, if `_exit_status` is 1, the command is run.
- Asynchronous publishing of CLI events--The CLI event is published, and then the CLI command is executed.
- Asynchronous publishing of CLI events with command skipping--The CLI event is published, but the CLI command is not executed.

### Counter Event Detector

The counter event detector publishes an event when a named counter crosses a specified threshold. There are two or more participants that affect counter processing. The counter event detector can modify the counter, and one or more subscribers define the criteria that cause the event to be published. After a counter event has been published, the counter monitoring logic can be reset to start monitoring the counter immediately or it can be reset when a second threshold--called an exit value--is crossed.

### Custom CLI Event Detector

The custom CLI event detector publishes an event to add and enhance existing CLI command syntax. When the special parser characters Tab, ? (question mark), and Enter are entered, the parser sends the input to the custom CLI event detector for processing. The custom CLI event detector then compares this input against registered strings to determine if this is a new or enhanced CLI command. Upon a match the custom CLI event detector takes appropriate actions, such as displaying help for the command if ? is entered, displaying the entire command if Tab is entered, or executing the command if Enter was entered. If a match does not occur, the parser regains control and processes the information as usual.

### Enhanced Object Tracking Event Detector

The enhanced object tracking (EOT) event detector publishes an event when the status of a tracked object changes. Object tracking was first introduced into the Hot Standby Router Protocol (HSRP) as a simple tracking mechanism that allowed you to track the interface line-protocol state only. If the line-protocol state of the interface went down, the HSRP priority of the device was reduced, allowing another HSRP device with a higher priority to become active.

Object tracking was enhanced to provide complete separation between the objects to be tracked and the action to be taken by a client when a tracked object changes. Thus, several clients such as HSRP, VRRP, or GLBP can register their interest with the tracking process, track the same object, and each take different action when the object changes. Each tracked object is identified by a unique number that is specified on the tracking command-line interface (CLI). Client processes use this number to track a specific object. The tracking process periodically polls the tracked objects and notes any change of value. The changes in the tracked object are communicated to interested client processes, either immediately or after a specified delay. The object values are reported as either up or down.

Enhanced object tracking is now integrated with EEM to allow EEM to report on a status change of a tracked object and to allow enhanced object tracking to track EEM objects. A new type of tracking object--a stub object--is created. The stub object can be manipulated using the existing CLI commands that already allow tracked objects to be manipulated.

### GOLD Event Detector

The GOLD event detector publishes an event when a GOLD failure event is detected on a specified card and subcard.

### Interface Counter Event Detector

The interface counter event detector publishes an event when a generic Cisco IOS interface counter for a specified interface crosses a defined threshold. A threshold can be specified as an absolute value or an incremental value. If the incremental value is set to 50, for example, an event would be published when the interface counter increases by 50.

After an interface counter event has been published, the interface counter monitoring logic is reset using two methods. The interface counter is reset either when a second threshold--called an exit value--is crossed or when an elapsed period of time occurs.

### IP SLA Event Detector

The IP SLA event detector publishes an event when an IP SLA reaction is triggered.

### NetFlow Event Detector

The NetFlow event detector publishes an event when a NetFlow event is triggered.

### None Event Detector

The none event detector publishes an event when the Cisco IOS **event manager run** CLI command executes an EEM policy. EEM schedules and runs policies on the basis of an event specification that is contained within the policy itself. An EEM policy must be identified and registered to be permitted to run manually before the **event manager run** command will execute.



### OIR Event Detector

The online insertion and removal (OIR) event detector publishes an event when one of the following hardware insertion or removal event occurs:

- A card is removed.
- A card is inserted.

Route Processors (RPs), line cards, or feature cards can be monitored for OIR events.

### Resource Event Detector

The resource event detector publishes an event when the Embedded Resource Manager (ERM) reports an event for the specified policy. The ERM infrastructure tracks resource depletion and resource dependencies across processes and within a system to handle various error conditions. The error conditions are handled by providing an equitable sharing of resources between various applications. The ERM framework provides a communication mechanism for resource entities and allows communication between these resource entities from numerous locations. The ERM framework also helps in debugging CPU and memory-related issues. The ERM monitors system resource usage to better understand scalability needs by allowing you to configure threshold values for resources such as the CPU, buffers, and memory. The ERM event detector is the preferred method for monitoring resources in Cisco software but the ERM event detector is not supported in Software Modularity images. For more details about ERM, go to “Embedded Resource Manager” module.

### RF Event Detector

The redundancy framework (RF) event detector publishes an event when one or more RF events occur during synchronization in a dual Route Processor (RP) system. The RF event detector can also detect an event when a dual RP system continuously switches from one RP to another RP (referred to as a ping-pong situation).

### RPC Event Detector

The remote procedure call (RPC) event detector provides the ability to invoke EEM policies from outside the device over an encrypted connection using Secure Shell (SSH). The RPC event detector uses Simple Object Access Protocol (SOAP) data encoding for exchanging XML-based messages. This event detector can be used to run EEM policies and then receive output in a SOAP XML-formatted reply.

### Routing Event Detector

The routing event detector publishes an event when a route entry changes in the Routing Information Base (RIB).

### SNMP Event Detector

The SNMP event detector allows a standard SNMP MIB object to be monitored and an event to be generated when the object matches specified values or crosses specified thresholds.

### SNMP Notification Event Detector

The SNMP notification event detector provides the ability to intercept SNMP trap and inform messages coming into or going out of the device. An SNMP notification event is generated when an incoming or outgoing SNMP trap or inform message matches specified values or crosses specified thresholds. The SNMP event detector can wait and intercept the outgoing SNMP traps and informs.

### SNMP Object Event Detector

The Simple Network Management Protocol (SNMP) object trap event detector provides an extension to replace the value when an SNMP trap with the specified SNMP object ID (OID) is encountered on a specific interface or address.

### Syslog Event Detector

The syslog event detector allows for screening syslog messages for a regular expression pattern match. The selected messages can be further qualified, requiring that a specific number of occurrences be logged within a specified time. A match on a specified event criteria triggers a configured policy action.

### System Manager Event Detector

The system manager event detector generates events for Cisco IOS Software Modularity process start, normal or abnormal stop, and restart events. The events generated by the system manager allows policies to change the default behavior of the process restart.

### Timer Event Detector

The timer event detector publishes events for the following four different types of timers:

- An absolute-time-of-day timer publishes an event when a specified absolute date and time occurs.
- A countdown timer publishes an event when a timer counts down to zero.
- A watchdog timer publishes an event when a timer counts down to zero and then the timer automatically resets itself to its initial value and starts to count down again.
- A CRON timer publishes an event using a UNIX standard CRON specification to indicate when the event is to be published. A CRON timer never publishes events more than once per minute.

### Watchdog System Monitor (IOSWDSysMon) Event Detector for Cisco IOS

The Cisco IOS watchdog system monitor event detector publishes an event when one of the following occurs:

- CPU utilization for a Cisco IOS task crosses a threshold.
- Memory utilization for a Cisco IOS task crosses a threshold.



---

**Note** Cisco IOS processes are now referred to as tasks to distinguish them from Cisco IOS Software Modularity processes.

---

Two events may be monitored at the same time, and the event publishing criteria can be specified to require one event or both events to cross their specified thresholds.

### Watchdog System Monitor (WDSysMon) Event Detector for Cisco IOS Software Modularity

The Cisco IOS Software Modularity watchdog system monitor event detector detects infinite loops, deadlocks, and memory leaks in Cisco IOS Software Modularity processes.

## EEM Actions Available by Cisco IOS Release

The CLI-based corrective actions that are taken when event detectors report events enable a powerful on-device event management mechanism. Some actions are available in every Cisco IOS release, but most actions have been introduced in a specific release. Use the table below to determine which actions are available in your specific Cisco IOS release. A blank entry (--) indicates that the action is not available; the text “Yes” indicates that the action is available. The actions shown in the table are supported in later releases of the same Cisco IOS release train. For more details on each action, see the Embedded Event Manager Actions concept in the “Embedded Event Manager Overview” module.

**Table 194: Availability of Actions by Cisco IOS Release**

| Action                                | 12.2(25)S | 12.3(14)T<br>12.2(18)SXF5<br>12.2(28)SB<br>12.2(33)SRA | 12.4(2)T<br>12.2(31)SB3<br>12.2(33)SRB | 12.2(18)SXF4<br>Cisco IOS<br>Software<br>Modularity | 12.2(33)SXH | 12.4(20)T | 12.4(22)T | 15.0(1)M | 15E<br>XE 3E |
|---------------------------------------|-----------|--------------------------------------------------------|----------------------------------------|-----------------------------------------------------|-------------|-----------|-----------|----------|--------------|
| Execute a CLI command                 | --        | Yes                                                    | Yes                                    | Yes                                                 | Yes         | Yes       | Yes       | Yes      | Yes          |
| Generate a CNS event                  | Yes       | Yes                                                    | Yes                                    | Yes                                                 | Yes         | Yes       | Yes       | Yes      | Yes          |
| Generate a prioritized syslog message | Yes       | Yes                                                    | Yes                                    | Yes                                                 | Yes         | Yes       | Yes       | Yes      | Yes          |
| Generate an SNMP trap                 | Yes       | Yes                                                    | Yes                                    | Yes                                                 | Yes         | Yes       | Yes       | Yes      | Yes          |
| Manually run an EEM policy            | --        | Yes                                                    | Yes                                    | Yes                                                 | Yes         | Yes       | Yes       | Yes      | Yes          |
| Publish an application-specific event | Yes       | Yes                                                    | Yes                                    | Yes                                                 | Yes         | Yes       | Yes       | Yes      | Yes          |
| Read the state of a tracked object    | --        | --                                                     | Yes                                    | --                                                  |             | Yes       | Yes       | Yes      | Yes          |
| Reload the Cisco software             | Yes       | Yes                                                    | Yes                                    | Yes                                                 | Yes         | Yes       | Yes       | Yes      | Yes          |
| Request system information            | --        | Yes                                                    | Yes                                    | Yes                                                 | Yes         | Yes       | Yes       | Yes      | Yes          |
| Send a short e-mail                   | --        | Yes                                                    | Yes                                    | Yes                                                 | Yes         | Yes       | Yes       | Yes      | Yes          |
| Set or modify a named counter         | Yes       | Yes                                                    | Yes                                    | Yes                                                 | Yes         | Yes       | Yes       | Yes      | Yes          |
| Set the state of a tracked object     | --        | --                                                     | Yes                                    | --                                                  |             | Yes       | Yes       | Yes      | Yes          |
| Switch to a secondary RP              | Yes       | Yes                                                    | Yes                                    | Yes                                                 | Yes         | Yes       | Yes       | Yes      | Yes          |

## Embedded Event Manager Actions

The CLI-based corrective actions that are taken when event detectors report events enable a powerful on-device event management mechanism. Some EEM actions are available on every Cisco IOS release, but most EEM actions have been introduced in a specific release. For details of which EEM action is supported in each Cisco IOS release, see the EEM Actions Available by Cisco IOS Release concept in the “Writing Embedded Event Manager Policies Using the Cisco IOS CLI” or the “Writing Embedded Event Manager Policies Using Tcl” modules. EEM supports the following actions:

- Executing a Cisco IOS command-line interface (CLI) command.
- Generating a CNS event for upstream processing by Cisco CNS devices.
- Setting or modifying a named counter.
- Switching to a secondary processor in a fully redundant hardware configuration.
- Requesting system information when an event occurs.
- Sending a short e-mail.
- Manually running an EEM policy.
- Publishing an application-specific event.
- Reloading the Cisco software.
- Generating an SNMP trap.
- Generating prioritized syslog messages.
- Reading the state of a tracked object.
- Setting the state of a tracked object.

EEM action CLI commands contain an EEM action label that is a unique identifier that can be any string value. Actions are sorted and run in ascending alphanumeric (lexicographical) key sequence using the label as the sort key. If you are using numbers as labels be aware that alphabetical sorting will sort 10.0 after 1.0, but before 2.0, and in this situation we recommend that you use numbers such as 01.0, 02.0, and so on, or use an initial letter followed by numbers.

## Embedded Event Manager Environment Variables

EEM allows environment variables to be used in EEM policies. Tool Command Language (Tcl) allows global variables to be defined that are known to all procedures within a Tcl script. EEM allows environment variables to be defined using a CLI command, the **event manager environment** command, for use within an EEM policy. All EEM environment variables are automatically assigned to Tcl global variables before a Tcl script is run. There are three different types of environment variables associated with Embedded Event Manager:

- User-defined--Defined by you if you create an environment variable in a policy that you have written.
- Cisco-defined--Defined by Cisco for a specific sample policy.
- Cisco built-in (available in EEM applets)--Defined by Cisco and can be read only or read/write. The read only variables are set by the system before an applet starts to execute. The single read/write variable, `_exit_status`, allows you to set the exit status at policy exit for policies triggered from synchronous events.

Cisco-defined environment variables (see the table below) and Cisco system-defined environment variables may apply to one specific event detector or to all event detectors. Environment variables that are user-defined or defined by Cisco in a sample policy are set using the **event manager environment** command. Variables that are used in the EEM policy must be defined before you register the policy. A Tcl policy contains a section called “Environment Must Define” that can be defined to check that any required environment variables are defined before the policy runs.

Cisco built-in environment variables are a subset of the Cisco-defined environment variables and the built-in variables are available to EEM applets only. The built-in variables can be read-only or can be read and write, and these variables may apply to one specific event detector or to all event detectors. For more details and a table listing the Cisco system-defined variables, see the “Writing Embedded Event Manager Policies Using the Cisco IOS CLI” module.



**Note** Cisco-defined environment variables begin with an underscore character (\_). We strongly recommend that customers avoid the same naming convention to prevent naming conflicts.

The table below describes the Cisco-defined variables used in the sample EEM policies. Some of the environment variables do not have to be specified for the corresponding sample policy to run and these are marked as optional.

**Table 195: Cisco-Defined Environmental Variables and Examples**

| Environment Variable  | Description                                                                                                                                                                               | Example                                       |
|-----------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------|
| _config_cmd1          | The first configuration command that is executed.                                                                                                                                         | <b>interface Ethernet1/0</b>                  |
| _config_cmd2          | (Optional) The second configuration command that is executed.                                                                                                                             | <b>no shutdown</b>                            |
| _crash_reporter_debug | (Optional) A value that identifies whether debug information for tm_crash_reporter.tcl will be enabled.                                                                                   | 1                                             |
| _crash_reporter_url   | The URL location to which the crash report is sent.                                                                                                                                       | http://www.yourdomain.com/fm/interface_tm.cgi |
| _cron_entry           | A CRON specification that determines when the policy will run. See the “Writing Embedded Event Manager Policies Using Tcl” module for more information about how to specify a cron entry. | 0-59/1 0-23/1 * * 0-7                         |
| _email_server         | A Simple Mail Transfer Protocol (SMTP) mail server used to send e-mail.                                                                                                                   | mailserver.yourdomain.com                     |
| _email_to             | The address to which e-mail is sent.                                                                                                                                                      | engineer@yourdomain.com                       |
| _email_from           | The address from which e-mail is sent.                                                                                                                                                    | devtest@yourdomain.com                        |
| _email_cc             | The address to which the e-mail is be copied.                                                                                                                                             | manager@yourdomain.com                        |

| Environment Variable     | Description                                                                                                                                                                                                                                 | Example                                                |
|--------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------|
| _email_ipaddr            | The source IP address of the recipient.                                                                                                                                                                                                     | 209.165.201.1 or (IPv6 address)<br>2001:0DB8::1        |
| _info_snmp_oid           | The SNMP object ID.                                                                                                                                                                                                                         | 1.3.6.1.2.1.2 or<br>iso.internet.mgmt.mib-2.interfaces |
| _info_snmp_value         | The value string of the associated SNMP data element.                                                                                                                                                                                       |                                                        |
| _show_cmd                | The CLI <b>show</b> command to be executed when the policy is run.                                                                                                                                                                          | <b>show version</b>                                    |
| _syslog_pattern          | A regular expression pattern match string that is used to compare syslog messages to determine when the policy runs.                                                                                                                        | .*UPDOWN.*FastEthernet 0/0.*                           |
| _tm_fsys_usage_cron      | (Optional) A CRON specification that is used in the <b>event_register</b> keyword extension. If unspecified, the <code>_tm_fsys_usage.tcl</code> policy is triggered once per minute.                                                       | 0-59/1 0-23/1 * * 0-7                                  |
| _tm_fsys_usage_debug     | (Optional) When this variable is set to a value of 1, disk usage information is displayed for all entries in the system.                                                                                                                    | 1                                                      |
| _tm_fsys_usage_freebytes | (Optional) Free byte threshold for systems or specific prefixes. If free space falls below a given value, a warning is displayed.                                                                                                           | disk2:98000000                                         |
| _tm_fsys_usage_percent   | (Optional) Disk usage percentage thresholds for systems or specific prefixes. If disk usage percentage exceeds a given percentage, a warning is displayed. If unspecified, the default disk usage percentage is 80 percent for all systems. | nvrnram:25 disk2:5                                     |

## Embedded Event Manager Policy Creation

EEM is a policy driven process in which the EEM policy engine receives notifications when faults and other events occur in the Cisco software system. Embedded Event Manager policies implement recovery based on the current state of the system and the actions specified in the policy for a given event. Recovery actions are triggered when the policy is run.

Although there are some EEM CLI configuration and **show** commands, EEM is implemented through the creation of policies. An EEM policy is an entity that defines an event and the actions to be taken when that event occurs. There are two types of EEM policies: an applet or a script. An applet is a simple form of policy that is defined within the CLI configuration. A script is a form of policy that is written in Tcl.

The creation of an EEM policy involves:

- Selecting the event for which the policy is run.

- Defining the event detector options associated with logging and responding to the event.
- Defining the environment variables, if required.
- Choosing the actions to be performed when the event occurs.

There are two ways to create an EEM policy. The first method is to write applets using CLI commands, and the second method is to write Tcl scripts. Cisco provides enhancements to Tcl in the form of Tcl command extensions that facilitate the development of EEM policies. Scripts are defined off the networking device using an ASCII editor. The script is then copied to the networking device and registered with EEM. When a policy is registered with the Embedded Event Manager, the software examines the policy and registers it to be run when the specified event occurs. Policies can be unregistered or suspended. Both types of policies can be used to implement EEM in your network.

For details on writing EEM policies using the Cisco IOS CLI, go to “Writing Embedded Event Manager Policies Using the Cisco IOS CLI” module.

For details on writing EEM policies using Tcl, go to “Writing Embedded Event Manager Policies Using Tcl” module.

## Where to Go Next

- If you want to write EEM policies using the Cisco IOS CLI, see the “Writing Embedded Event Manager Policies Using the Cisco IOS CLI” module.
- If you want to write EEM policies using Tcl, see the “Writing Embedded Event Manager Policies Using Tcl” module.

## Feature Information for Embedded Event Manager 4.0 Overview

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to [www.cisco.com/go/cfn](http://www.cisco.com/go/cfn). An account on Cisco.com is not required.

**Table 196: Feature Information for Embedded Event Manager 4.0 Overview**

| Feature Name               | Releases      | Feature Information                                           |
|----------------------------|---------------|---------------------------------------------------------------|
| Embedded Event Manager 4.0 | IOS 15.2(5)E1 | This feature was introduced and is supported only on c2960cx. |

## Additional References

The following sections provide references related to EEM.

**Related Documents**

| Related Topic                                                                                                  | Document Title                                                         |
|----------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------|
| EEM commands: complete command syntax, defaults, command mode, command history, usage guidelines, and examples | <a href="#">Cisco IOS Embedded Event Manager Command Reference</a>     |
| Embedded Event Manager policy writing using the CLI                                                            | Writing Embedded Event Manager Policies Using the Cisco IOS CLI module |
| Embedded Event Manager policy writing using Tcl                                                                | Writing Embedded Event Manager Policies Using Tcl module               |
| Embedded Resource Manager                                                                                      | Embedded Resource Manager module                                       |

**Standards**

| Standard                                                                                              | Title |
|-------------------------------------------------------------------------------------------------------|-------|
| No new or modified standards are supported, and support for existing standards has not been modified. | --    |

**MIBs**

| MIB                          | MIBs Link                                                                                                                                                                                                              |
|------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| CISCO-EMBEDDED-EVENT-MGR-MIB | To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL:<br><a href="http://www.cisco.com/go/mibs">http://www.cisco.com/go/mibs</a> |

**RFCs**

| RFC                                                                                         | Title |
|---------------------------------------------------------------------------------------------|-------|
| No new or modified RFCs are supported, and support for existing RFCs has not been modified. | --    |



**Technical Assistance**

| Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             | Link                                                                                                                     |
|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------|
| <p>The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies.</p> <p>To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds.</p> <p>Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.</p> | <p><a href="http://www.cisco.com/cisco/web/support/index.html">http://www.cisco.com/cisco/web/support/index.html</a></p> |





## CHAPTER 100

# Information About Writing EEM Policies Using the Cisco IOS CLI

---

- [Prerequisites for Writing EEM Policies Using the Cisco IOS CLI, on page 2001](#)
- [Information About Writing EEM Policies Using the Cisco IOS CLI, on page 2001](#)
- [How to Write EEM Policies Using the Cisco IOS CLI, on page 2013](#)
- [Configuration Examples for Writing Embedded Event Manager Policies Using Tcl, on page 2056](#)
- [Additional References, on page 2072](#)
- [Feature Information for Writing EEM 4.0 Policies Using the Cisco IOS CLI, on page 2074](#)

## Prerequisites for Writing EEM Policies Using the Cisco IOS CLI

- Before writing EEM policies, you should be familiar with the concepts explained in the “Embedded Event Manager Overview” module.
- If the **action cns-event** command is used, access to a Cisco Networking Services (CNS) Event gateway must be configured.
- If the **action force-switchover** command is used, a secondary processor must be configured on the device.
- If the **action snmp-trap** command is used, the **snmp-server enable traps event-manager** command must be enabled to permit SNMP traps to be sent from the Cisco IOS device to the SNMP server. Other relevant **snmp-server** commands must also be configured; for details see the **action snmp-trap** command page.

## Information About Writing EEM Policies Using the Cisco IOS CLI

### Embedded Event Manager Policies

EEM offers the ability to monitor events and take informational or corrective action when the monitored events occur or a threshold is reached. An EEM policy is an entity that defines an event and the actions to be

taken when that event occurs. There are two types of EEM policies: an applet or a script. An applet is a simple form of policy that is defined within the CLI configuration. A script is a form of policy that is written in Tool Command Language (Tcl).

## EEM Applet

An EEM applet is a concise method for defining event screening criteria and the actions to be taken when that event occurs. In applet configuration mode, three types of configuration statements are supported. The **event** commands are used to specify the event criteria to trigger the applet to run, the **action** commands are used to specify an action to perform when the EEM applet is triggered, and the **set** command is used to set the value of an EEM applet variable. Currently only the `_exit_status` variable is supported for the **set** command.

Only one **event** configuration command is allowed within an applet configuration. When applet configuration mode is exited and no **event** command is present, a warning is displayed stating that no event is associated with this applet. If no event is specified, this applet is not considered registered. When no action is associated with this applet, events are still triggered but no actions are performed. Multiple **action** configuration commands are allowed within an applet configuration. Use the **show event manager policy registered** command to display a list of registered applets.

Before modifying an EEM applet, be aware that the existing applet is not replaced until you exit applet configuration mode. While you are in applet configuration mode modifying the applet, the existing applet may be executing. It is safe to modify the applet without unregistering it. When you exit applet configuration mode, the old applet is unregistered and the new version is registered.

The action configuration commands are uniquely identified using the *label* argument, which can be any string value. Actions are sorted in ascending alphanumeric key sequence using the *label* argument as the sort key, and they are run using this sequence.

The Embedded Event Manager schedules and runs policies on the basis of an event specification that is contained within the policy itself. When applet configuration mode is exited, EEM examines the **event** and **action** commands that are entered and registers the applet to be run when a specified event occurs.

## EEM Script

Scripts are defined off the networking device using an ASCII editor. The script is then copied to the networking device and registered with EEM. Tcl scripts are supported by EEM.

EEM allows you to write and implement your own policies using Tcl. Writing an EEM policy involves:

- Selecting the event for which the policy is run.
- Defining the event detector options associated with logging and responding to the event.
- Choosing the actions to be followed when the event occurs.

Cisco provides enhancements to Tcl in the form of keyword extensions that facilitate the development of EEM policies. The main categories of keywords identify the detected event, the subsequent action, utility information, counter values, and system information. For more details about writing EEM policies using Tcl, see the “Writing Embedded Event Manager Policies Using Tcl” module.

## Embedded Event Manager Built-In Environment Variables Used in EEM Applets

EEM built-in environment variables are a subset of the Cisco-defined environment variables and the built-in variables are available to EEM applets only. The built-in variables can be read-only or can be read and write

and these variables may apply to one specific event detector or to all event detectors. The table below lists the Cisco built-in environment variables that are read-only alphabetically by event detector and subevent.

**Table 197: EEM Built-In Environment Variables (Read Only)**

| <b>Environment Variable</b>                  | <b>Description</b>                                                                                                                                           |
|----------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------|
| All Events                                   |                                                                                                                                                              |
| <b>_event_id</b>                             | Unique number that indicates the ID for this published event. Multiple policies may be run for the same event, and each policy will have the same event_id.  |
| <b>_event_type</b>                           | Type of event.                                                                                                                                               |
| <b>_event_type_string</b>                    | An ASCII string identifier of the event type that triggered the event.                                                                                       |
| <b>_event_pub_sec</b> <b>_event_pub_msec</b> | The time, in seconds and milliseconds, at which the event was published to the EEM.                                                                          |
| <b>_event_severity</b>                       | The severity of the event.                                                                                                                                   |
| Application-Specific Event Detector          |                                                                                                                                                              |
| <b>_application_component_id</b>             | The event application component identifier.                                                                                                                  |
| <b>_application_data1</b>                    | The value of an environment variable, character text, or a combination of the two to be passed to an application-specific event when the event is published. |
| <b>_application_data2</b>                    | The value of an environment variable, character text, or a combination of the two to be passed to an application-specific event when the event is published. |
| <b>_application_data3</b>                    | The value of an environment variable, character text, or a combination of the two to be passed to an application-specific event when the event is published. |
| <b>_application_data4</b>                    | The value of an environment variable, character text, or a combination of the two to be passed to an application-specific event when the event is published. |
| <b>_application_sub_system</b>               | The event application subsystem number.                                                                                                                      |
| <b>_application_type</b>                     | The type of application.                                                                                                                                     |
| CLI Event Detector                           |                                                                                                                                                              |
| <b>_cli_msg</b>                              | The fully expanded message that triggered the CLI event.                                                                                                     |
| <b>_cli_msg_count</b>                        | The number of times that a message match occurred before the event was published.                                                                            |
| Counter Event Detector                       |                                                                                                                                                              |

| Environment Variable                    | Description                                                                                                                                                                                                                                            |
|-----------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>_counter_name</b>                    | The name of the counter.                                                                                                                                                                                                                               |
| <b>_counter_value</b>                   | The value of the counter.                                                                                                                                                                                                                              |
| Enhanced Object Tracking Event Detector |                                                                                                                                                                                                                                                        |
| <b>_track_number</b>                    | The number of the tracked object.                                                                                                                                                                                                                      |
| <b>_track_state</b>                     | The state of the tracked object; down or up.                                                                                                                                                                                                           |
| GOLD Event Detector                     |                                                                                                                                                                                                                                                        |
| <b>_action_notify</b>                   | The action notify information in a GOLD event flag; either false or true.                                                                                                                                                                              |
| <b>_event_severity</b>                  | The event severity which can be one of the following; normal, minor, or major.                                                                                                                                                                         |
| <b>_gold_bl</b>                         | The boot diagnostic level, which can be one of the following values: <ul style="list-style-type: none"> <li>• 0: complete diagnostic</li> <li>• 1: minimal diagnostic</li> <li>• 2: bypass diagnostic</li> </ul>                                       |
| <b>_gold_card</b>                       | The card on which a GOLD failure event was detected.                                                                                                                                                                                                   |
| <b>_gold_cf</b> <i>testnum</i>          | Consecutive failure, where <i>testnum</i> is the test number. For example, <b>_gold_cf3</b> is the EEM built-in environment variable for consecutive failure of test 3.                                                                                |
| <b>_gold_ci</b>                         | Card index.                                                                                                                                                                                                                                            |
| <b>_gold_cn</b>                         | Card name.                                                                                                                                                                                                                                             |
| <b>_gold_ec</b> <i>testnum</i>          | Test error code, where <i>testnum</i> is the test number. For example, <b>_gold_ec3</b> is the EEM built-in environment variable for the error code of test 3.                                                                                         |
| <b>_gold_lf</b> <i>testnum</i>          | Last fail time, where <i>testnum</i> is the test number. For example, <b>_gold_lf3</b> is the EEM built-in variable for the last fail time of test 3.<br><br>The time-stamp format is <i>mmm dd yyyy hh:mm:ss</i> . For example, Mar 11 2005 08:47:00. |
| <b>_gold_new_failure</b>                | The new test failure information in a GOLD event flag; either true or false.                                                                                                                                                                           |

| Environment Variable           | Description                                                                                                                                                                                                                                                                                                                                                                                                   |
|--------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>_gold_overall_result</b>    | The overall diagnostic result, which can be one of the following values: <ul style="list-style-type: none"> <li>• 0: OK</li> <li>• 3: minor error</li> <li>• 4: major error</li> <li>• 14: unknown result</li> </ul>                                                                                                                                                                                          |
| <b>_gold_pc</b>                | Port counts.                                                                                                                                                                                                                                                                                                                                                                                                  |
| <b>_gold_rc</b> <i>testnum</i> | Test total run count, where <i>testnum</i> is the test number. For example, <b>_gold_rc3</b> is the EEM built-in variable for the total run count of test 3.                                                                                                                                                                                                                                                  |
| <b>_gold_sn</b>                | Card serial number.                                                                                                                                                                                                                                                                                                                                                                                           |
| <b>_gold_sub_card</b>          | The subcard on which a GOLD failure event was detected.                                                                                                                                                                                                                                                                                                                                                       |
| <b>_gold_ta</b> <i>testnum</i> | Test attribute, where <i>testnum</i> is the test number. For example, <b>_gold_ta3</b> is the EEM built-in variable for the test attribute of test 3.                                                                                                                                                                                                                                                         |
| <b>_gold_tc</b>                | Test counts.                                                                                                                                                                                                                                                                                                                                                                                                  |
| <b>_gold_tf</b> <i>testnum</i> | Total failure count, where <i>testnum</i> is the test number. For example, <b>_gold_tf3</b> is the EEM built-in variable for the total failure count of test 3.                                                                                                                                                                                                                                               |
| <b>_gold_tn</b> <i>testnum</i> | Test name, where <i>testnum</i> is the test number. For example, <b>_gold_tn3</b> is the EEM built-in variable for the name of test 3.                                                                                                                                                                                                                                                                        |
| <b>_gold_tr</b> <i>testnum</i> | Test result, where <i>testnum</i> is the test number. For example, <b>_gold_tr6</b> is the EEM built-in variable for test 6, where test 6 is not a per-port test and not a per-device test.<br><br>The test result is one of the following values: <ul style="list-style-type: none"> <li>• P: diagnostic result Pass</li> <li>• F: diagnostic result Fail</li> <li>• U: diagnostic result Unknown</li> </ul> |

| Environment Variable                     | Description                                                                                                                                                                                                                                                                                                                                                                                                                        |
|------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>_gold_tr</b> <i>testnum d devnum</i>  | Per-device test result, where <i>testnum</i> is the test number and <i>devnum</i> is the device number. For example, <b>_gold_tr3d20</b> is the EEM built-in variable for the test result for test 3, device 20.<br><br>The test result is one of the following values: <ul style="list-style-type: none"> <li>• P: diagnostic result Pass</li> <li>• F: diagnostic result Fail</li> <li>• U: diagnostic result Unknown</li> </ul> |
| <b>_gold_tr</b> <i>testnum p portnum</i> | Per-port test result, where <i>testnum</i> is the test number and <i>portnum</i> is the port number. For example, <b>_gold_tr5p20</b> is the EEM built-in variable for the test result for test 5, port 20.<br><br>The test result is one of the following values: <ul style="list-style-type: none"> <li>• P: diagnostic result Pass</li> <li>• F: diagnostic result Fail</li> <li>• U: diagnostic result Unknown</li> </ul>      |
| <b>_gold_tt</b>                          | The testing type, which can be one of the following: <ul style="list-style-type: none"> <li>• 1: a boot diagnostic</li> <li>• 2: an on-demand diagnostic</li> <li>• 3: a schedule diagnostic</li> <li>• 4: a monitoring diagnostic</li> </ul>                                                                                                                                                                                      |
| Interface Counter Event Detector         |                                                                                                                                                                                                                                                                                                                                                                                                                                    |
| <b>_interface_is_increment</b>           | A value to indicate whether the current interface counter value is an absolute value (0) or an increment value (1).                                                                                                                                                                                                                                                                                                                |
| <b>_interface_name</b>                   | The name of the interface to be monitored.                                                                                                                                                                                                                                                                                                                                                                                         |
| <b>_interface_parameter</b>              | The name of the interface counter to be monitored.                                                                                                                                                                                                                                                                                                                                                                                 |
| <b>_interface_value</b>                  | A value with which the current interface counter value is compared.                                                                                                                                                                                                                                                                                                                                                                |
| None Event Detector                      |                                                                                                                                                                                                                                                                                                                                                                                                                                    |
| <b>_event_id</b>                         | A value of 1 indicates an insertion event; a value of 2 indicates a removal event.                                                                                                                                                                                                                                                                                                                                                 |



| Environment Variable                                                                                                                                                                                                              | Description                                                                                       |
|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------|
| _none_arg<br>_none_arg1<br>_none_arg2<br>_none_arg3<br>_none_arg4<br>_none_arg5<br>_none_arg6<br>_none_arg7<br>_none_arg8<br>_none_arg9<br>_none_arg10<br>_none_arg11<br>_none_arg12<br>_none_arg13<br>_none_arg14<br>_none_arg15 | The parameters that are passed from the XML SOAP command to the script.                           |
| OIR Event Detector                                                                                                                                                                                                                |                                                                                                   |
| _oir_event                                                                                                                                                                                                                        | A value of 1 indicates an insertion event; a value of 2 indicates a removal event.                |
| _oir_slot                                                                                                                                                                                                                         | The slot number for the OIR event.                                                                |
| Resource Event Detector                                                                                                                                                                                                           |                                                                                                   |
| _resource_configured_threshold                                                                                                                                                                                                    | The configured ERM threshold.                                                                     |
| _resource_current_value                                                                                                                                                                                                           | The current value reported by ERM.                                                                |
| _resource_dampen_time                                                                                                                                                                                                             | The ERM dampen time, in nanoseconds.                                                              |
| _resource_direction                                                                                                                                                                                                               | The ERM event direction. The event direction can be one of the following: up, down, or no change. |
| _resource_level                                                                                                                                                                                                                   | The ERM event level. The four event levels are normal, minor, major, and critical.                |
| _resource_notify_data_flag                                                                                                                                                                                                        | The ERM notify data flag.                                                                         |
| _resource_owner_id                                                                                                                                                                                                                | The ERM resource owner ID.                                                                        |
| _resource_policy_id                                                                                                                                                                                                               | The ERM policy ID.                                                                                |

| Environment Variable                                                                                                                                                                                                                                                                                                                                                                                                              | Description                                                                                                                        |
|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------|
| <code>_resource_policy_violation_flag</code>                                                                                                                                                                                                                                                                                                                                                                                      | The ERM policy violation flag; either false or true.                                                                               |
| <code>_resource_time_sent</code>                                                                                                                                                                                                                                                                                                                                                                                                  | The ERM event time, in nanoseconds.                                                                                                |
| <code>_resource_user_id</code>                                                                                                                                                                                                                                                                                                                                                                                                    | The ERM resource user ID.                                                                                                          |
| RF Event Detector                                                                                                                                                                                                                                                                                                                                                                                                                 |                                                                                                                                    |
| <code>_rf_event</code>                                                                                                                                                                                                                                                                                                                                                                                                            | A value of 0 indicates that this is not an RF event; a value of 1 indicates an RF event.                                           |
| RPC Event Detector                                                                                                                                                                                                                                                                                                                                                                                                                |                                                                                                                                    |
| <code>_rpc_event</code>                                                                                                                                                                                                                                                                                                                                                                                                           | A value of 0 indicates that there is no error; a value of 1 to 83 indicates error.                                                 |
| <code>_rpc_argc</code><br><code>_rpc_arg0</code><br><code>_rpc_arg1</code><br><code>_rpc_arg2</code><br><code>_rpc_arg3</code><br><code>_rpc_arg4</code><br><code>_rpc_arg5</code><br><code>_rpc_arg6</code><br><code>_rpc_arg7</code><br><code>_rpc_arg8</code><br><code>_rpc_arg9</code><br><code>_rpc_arg10</code><br><code>_rpc_arg11</code><br><code>_rpc_arg12</code><br><code>_rpc_arg13</code><br><code>_rpc_arg14</code> | The parameters that are passed from the XML SOAP command to the applet.                                                            |
| SNMP Event Detector                                                                                                                                                                                                                                                                                                                                                                                                               |                                                                                                                                    |
| <code>_snmp_exit_event</code>                                                                                                                                                                                                                                                                                                                                                                                                     | A value of 0 indicates that this is not an exit event; a value of 1 indicates an exit event.                                       |
| <code>_snmp_oid</code>                                                                                                                                                                                                                                                                                                                                                                                                            | The SNMP object ID that caused the event to be published.                                                                          |
| <code>_snmp_oid_delta_val</code>                                                                                                                                                                                                                                                                                                                                                                                                  | The actual incremental difference between the value of the current SNMP object ID and the value when the event was last triggered. |

| Environment Variable                    | Description                                                                                                                |
|-----------------------------------------|----------------------------------------------------------------------------------------------------------------------------|
| <b>_snmp_oid_val</b>                    | The SNMP object ID value when the event was published.                                                                     |
| SNMP Notification Event Detector        |                                                                                                                            |
| <b>_snmp_notif_oid</b>                  | A user specified object ID.                                                                                                |
| <b>_snmp_notif_oid_val</b>              | A user specified object ID value.                                                                                          |
| <b>_snmp_notif_src_ip_addr</b>          | The source IP address of the SNMP Protocol Data Unit (PDU).                                                                |
| <b>_snmp_notif_dest_ip_addr</b>         | The destination IP address of the SNMP PDU.                                                                                |
| <b>_x_x_x_x_x_x_x(varbinds)</b>         | The SNMP PDU varbind information.                                                                                          |
| <b>_snmp_notif_trunc_vb_buf</b>         | Indicates whether the varbind information has been truncated due to the lack of space in the buffer.                       |
| Syslog Event Detector                   |                                                                                                                            |
| <b>_syslog_msg</b>                      | The syslog message that caused the event to be published.                                                                  |
| System Manager (Process) Event Detector |                                                                                                                            |
| <b>_process_dump_count</b>              | The number of times that a Posix process was dumped.                                                                       |
| <b>_process_exit_status</b>             | The status of the Posix process at exit.                                                                                   |
| <b>_process_fail_count</b>              | The number of times that a Posix process failed.                                                                           |
| <b>_process_instance</b>                | The instance number of the Posix process.                                                                                  |
| <b>_process_last_respawn</b>            | The Posix process that was last respawned.                                                                                 |
| <b>_process_node_name</b>               | The node name of the Posix process.                                                                                        |
| <b>_process_path</b>                    | The path of the Posix process.                                                                                             |
| <b>_process_process_name</b>            | The name of the Posix process.                                                                                             |
| <b>_process_respawn_count</b>           | The number of times that a Posix process was respawned.                                                                    |
| Timer Event Detector                    |                                                                                                                            |
| <b>_timer_remain</b>                    | The time available before the timer expires.<br><b>Note</b> This environment variable is not available for the CRON timer. |
| <b>_timer_time</b>                      | The time at which the last event was triggered.                                                                            |
| <b>_timer_type</b>                      | The type of timer.                                                                                                         |

| Environment Variable                                                     | Description                                                                                                                                                                                                         |
|--------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Watchdog System Monitor (IOSWDSysMon)<br>Event Detector                  |                                                                                                                                                                                                                     |
| <b>_ioswd_node</b>                                                       | The slot number for the Route Processor (RP) reporting node.                                                                                                                                                        |
| <b>_ioswd_num_subs</b>                                                   | The number of subevents present.                                                                                                                                                                                    |
| All Watchdog System Monitor (IOSWDSysMon) Subevents                      |                                                                                                                                                                                                                     |
| <b>_ioswd_sub1_present</b><br><b>_ioswd_sub2_present</b>                 | A value to indicate whether subevent 1 or subevent 2 is present. A value of 1 means that the subevent is present; a value of 0 means that the subevent is not present.                                              |
| <b>_ioswd_sub1_type</b> <b>_ioswd_sub2_type</b>                          | The event type, either <code>cpu_proc</code> or <code>mem_proc</code> .                                                                                                                                             |
| Watchdog System Monitor (IOSWDSysMon)<br><code>cpu_proc</code> Subevents |                                                                                                                                                                                                                     |
| <b>_ioswd_sub1_path</b> <b>_ioswd_sub2_path</b>                          | A process name of subevents.                                                                                                                                                                                        |
| <b>_ioswd_sub1_period</b> <b>_ioswd_sub2_period</b>                      | The time period, in seconds and optional milliseconds, used for measurement in subevents.                                                                                                                           |
| <b>_ioswd_sub1_pid</b> <b>_ioswd_sub2_pid</b>                            | The process identifier of subevents.                                                                                                                                                                                |
| <b>_ioswd_sub1_taskname</b><br><b>_ioswd_sub2_taskname</b>               | The task name of subevents.                                                                                                                                                                                         |
| <b>_ioswd_sub1_value</b> <b>_ioswd_sub2_value</b>                        | The CPU utilization of subevents measured as a percentage.                                                                                                                                                          |
| Watchdog System Monitor (IOSWDSysMon)<br><code>mem_proc</code> Subevents |                                                                                                                                                                                                                     |
| <b>_ioswd_sub1_diff</b> <b>_ioswd_sub2_diff</b>                          | A percentage value of the difference that triggered the event.<br><br><b>Note</b> This variable is set only when the <b>_ioswd_sub1_is_percent</b> or <b>_ioswd_sub2_is_percent</b> variable contains a value of 1. |
| <b>_ioswd_sub1_is_percent</b><br><b>_ioswd_sub2_is_percent</b>           | A number that identifies whether the value is a percentage. A value of 0 means that the value is not a percentage; a value of 1 means that the value is a percentage.                                               |
| <b>_ioswd_sub1_path</b> <b>_ioswd_sub2_path</b>                          | The process name of subevents.                                                                                                                                                                                      |
| <b>_ioswd_sub1_pid</b> <b>_ioswd_sub2_pid</b>                            | The process identifier of subevents.                                                                                                                                                                                |
| <b>_ioswd_sub1_taskname</b><br><b>_ioswd_sub2_taskname</b>               | The task name of subevents.                                                                                                                                                                                         |
| <b>_ioswd_sub1_value</b> <b>_ioswd_sub2_value</b>                        | The CPU utilization of subevents measured as a percentage.                                                                                                                                                          |

| Environment Variable                                                       | Description                                                                                                                                                            |
|----------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Watchdog System Monitor (WDSysMon)<br>Event Detector                       |                                                                                                                                                                        |
| <b>_wd_sub1_present _wd_sub2_present</b>                                   | A value to indicate whether subevent 1 or subevent 2 is present. A value of 1 means that the subevent is present; a value of 0 means that the subevent is not present. |
| <b>_wd_num_subs</b>                                                        | The number of subevents present.                                                                                                                                       |
| <b>_wd_sub1_type _wd_sub2_type</b>                                         | The event type: cpu_proc, cpu_tot, deadlock, dispatch_mgr, mem_proc, mem_tot_avail, or mem_tot_used.                                                                   |
| Watchdog System Monitor (WDSysMon)<br>cpu_proc Subevents                   |                                                                                                                                                                        |
| <b>_wd_sub1_node _wd_sub2_node</b>                                         | The slot number for the subevent RP reporting node.                                                                                                                    |
| <b>_wd_sub1_period _wd_sub2_period</b>                                     | The time period, in seconds and optional milliseconds, used for measurement in subevents.                                                                              |
| <b>_wd_sub1_procname _wd_sub2_procname</b>                                 | The process name of subevents.                                                                                                                                         |
| <b>_wd_sub1_value _wd_sub2_value</b>                                       | The CPU utilization of subevents measured as a percentage.                                                                                                             |
| Watchdog System Monitor (WDSysMon)<br>cpu_tot Subevents                    |                                                                                                                                                                        |
| <b>_wd_sub1_node _wd_sub2_node</b>                                         | The slot number for the subevent RP reporting node.                                                                                                                    |
| <b>_wd_sub1_period _wd_sub2_period</b>                                     | The time period, in seconds and optional milliseconds, used for measurement in subevents.                                                                              |
| <b>_wd_sub1_value _wd_sub2_value</b>                                       | The CPU utilization of subevents measured as a percentage.                                                                                                             |
| Watchdog System Monitor (WDSysMon)<br>deadlock Subevents                   |                                                                                                                                                                        |
| <b>_wd_sub1_entry_[1-N]_b_node<br/>_wd_sub2_entry_[1-N]_b_node</b>         | The slot number for the subevent RP reporting node.                                                                                                                    |
| <b>_wd_sub1_entry_[1-N]_b_pid<br/>_wd_sub2_entry_[1-N]_b_pid</b>           | The process identifier of subevents.                                                                                                                                   |
| <b>_wd_sub1_entry_[1-N]_b_procname<br/>_wd_sub2_entry_[1-N]_b_procname</b> | The process name of subevents.                                                                                                                                         |
| <b>_wd_sub1_entry_[1-N]_b_tid<br/>_wd_sub2_entry_[1-N]_b_tid</b>           | The time identifier of subevents.                                                                                                                                      |
| <b>_wd_sub1_entry_[1-N]_node<br/>_wd_sub2_entry_[1-N]_node</b>             | The slot number for the subevent RP reporting node.                                                                                                                    |

| Environment Variable                                                                     | Description                                                                                                                                                                                                               |
|------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <code>_wd_sub1_entry_[1-N]_pid</code><br><code>_wd_sub2_entry_[1-N]_pid</code>           | The process identifier of subevents.                                                                                                                                                                                      |
| <code>_wd_sub1_entry_[1-N]_procname</code><br><code>_wd_sub2_entry_[1-N]_procname</code> | The process name of subevents.                                                                                                                                                                                            |
| <code>_wd_sub1_entry_[1-N]_state</code><br><code>_wd_sub2_entry_[1-N]_state</code>       | The time identifier of subevents.                                                                                                                                                                                         |
| <code>_wd_sub1_entry_[1-N]_tid</code><br><code>_wd_sub2_entry_[1-N]_tid</code>           | The time identifier of subevents.                                                                                                                                                                                         |
| <code>_wd_sub1_num_entries</code><br><code>_wd_sub2_num_entries</code>                   | The number of subevents.                                                                                                                                                                                                  |
| Watchdog System Monitor (WDSysMon)<br>dispatch manager Subevents                         |                                                                                                                                                                                                                           |
| <code>_wd_sub1_node</code> <code>_wd_sub2_node</code>                                    | The slot number for the subevent RP reporting node.                                                                                                                                                                       |
| <code>_wd_sub1_period</code> <code>_wd_sub2_period</code>                                | The time period, in seconds and optional milliseconds, used for measurement in subevents.                                                                                                                                 |
| <code>_wd_sub1_procname</code> <code>_wd_sub2_procname</code>                            | The process name of subevents.                                                                                                                                                                                            |
| <code>_wd_sub1_value</code> <code>_wd_sub2_value</code>                                  | The CPU utilization of subevents measured as a percentage.                                                                                                                                                                |
| Watchdog System Monitor (WDSysMon)<br>mem_proc Subevents                                 |                                                                                                                                                                                                                           |
| <code>_wd_sub1_diff</code> <code>_wd_sub2_diff</code>                                    | A percentage value of the difference that triggered the event.<br><br><b>Note</b> This variable is set only when the <code>_wd_sub1_is_percent</code> or <code>_wd_sub2_is_percent</code> variable contains a value of 1. |
| <code>_wd_sub1_is_percent</code><br><code>_wd_sub2_is_percent</code>                     | A number that identifies whether the value is a percentage. A value of 0 means that the value is not a percentage; a value of 1 means that the value is a percentage.                                                     |
| <code>_wd_sub1_node</code> <code>_wd_sub2_node</code>                                    | The slot number for the subevent RP reporting node.                                                                                                                                                                       |
| <code>_wd_sub1_period</code> <code>_wd_sub2_period</code>                                | The time period, in seconds and optional milliseconds, used for measurement in subevents.                                                                                                                                 |
| <code>_wd_sub1_pid</code> <code>_wd_sub2_pid</code>                                      | The process identifier of subevents.                                                                                                                                                                                      |
| <code>_wd_sub1_procname</code> <code>_wd_sub2_procname</code>                            | The process name of subevents.                                                                                                                                                                                            |
| <code>_wd_sub1_value</code> <code>_wd_sub2_value</code>                                  | The CPU utilization of subevents measured as a percentage.                                                                                                                                                                |
| Watchdog System Monitor (WDSysMon)<br>mem_tot_avail and mem_tot_used Subevents           |                                                                                                                                                                                                                           |

| Environment Variable                                                 | Description                                                                                                                                                                                                           |
|----------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <code>_wd_sub1_avail</code> <code>_wd_sub2_avail</code>              | The memory available for subevents.                                                                                                                                                                                   |
| <code>_wd_sub1_diff</code> <code>_wd_sub2_diff</code>                | A percentage value of the difference that triggered the event.<br><b>Note</b> This variable is set only when the <code>_wd_sub1_is_percent</code> or <code>_wd_sub2_is_percent</code> variable contains a value of 1. |
| <code>_wd_sub1_is_percent</code><br><code>_wd_sub2_is_percent</code> | A number that identifies whether the value is a percentage. A value of 0 means that the value is not a percentage; a value of 1 means that the value is a percentage.                                                 |
| <code>_wd_sub1_node</code> <code>_wd_sub2_node</code>                | The slot number for the subevent RP reporting node.                                                                                                                                                                   |
| <code>_wd_sub1_period</code> <code>_wd_sub2_period</code>            | The time period, in seconds and optional milliseconds, used for measurement in subevents.                                                                                                                             |
| <code>_wd_sub1_value</code> <code>_wd_sub2_value</code>              | The CPU utilization of subevents measured as a percentage.                                                                                                                                                            |
| <code>_wd_sub1_used</code> <code>_wd_sub2_used</code>                | The memory used by subevents.                                                                                                                                                                                         |

## How to Write EEM Policies Using the Cisco IOS CLI

### Registering and Defining an Embedded Event Manager Applet

Perform this task to register an applet with Embedded Event Manager and to define the EEM applet using the Cisco IOS CLI **event** and **action** commands. Only one **event** command is allowed in an EEM applet. Multiple **action** commands are permitted. If no **event** and no **action** commands are specified, the applet is removed when you exit configuration mode.

The SNMP event detector and the syslog **action** commands used in this task are just representing any event detector and **action** commands. For examples using other event detectors and **action** commands, see the [Embedded Event Manager Applet Configuration Examples, on page 2056](#).

### EEM Environment Variables

EEM environment variables for EEM policies are defined using the EEM **event manager environment** configuration command. By convention, all Cisco EEM environment variables begin with “\_”. In order to avoid future conflict, customers are urged not to define new variables that start with “\_”.

You can display the EEM environment variables set on your system by using the **show event manager environment** privileged EXEC command.

For example, you can create EEM policies that can send e-mails when an event occurs. The table below describes the e-mail-specific environment variables that can be used in EEM policies.

Table 198: EEM E-mail-Specific Environmental Variables

| Environment Variable       | Description                                                             | Example                                                                                                                                                                                                                                                  |
|----------------------------|-------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <code>_email_server</code> | A Simple Mail Transfer Protocol (SMTP) mail server used to send e-mail. | The e-mail server name--Mailservername-- can be in any one of the following template formats: <ul style="list-style-type: none"> <li>• <code>username:password@host</code></li> <li>• <code>username@host</code></li> <li>• <code>host</code></li> </ul> |
| <code>_email_to</code>     | The address to which e-mail is sent.                                    | <code>engineering@example.com</code>                                                                                                                                                                                                                     |
| <code>_email_from</code>   | The address from which e-mail is sent.                                  | <code>devtest@example.com</code>                                                                                                                                                                                                                         |
| <code>_email_cc</code>     | The address to which the e-mail is copied.                              | <code>manager@example.com</code>                                                                                                                                                                                                                         |

## Alphabetical Order of EEM Action Labels

An EEM action label is a unique identifier that can be any string value. Actions are sorted and run in ascending alphanumeric (lexicographical) key sequence using the label as the sort key. If you are using numbers as labels be aware that alphanumerical sorting will sort 10.0 after 1.0, but before 2.0, and in this situation we recommend that you use numbers such as 01.0, 02.0, and so on, or use an initial letter followed by numbers.

### SUMMARY STEPS

1. **enable**
2. **show event manager environment** [**all**] *variable-name*
3. **configure terminal**
4. **event manager environment** *variable-name string*
5. Repeat [Alphabetical Order of EEM Action Labels](#) for all the required environment variables.
6. **event manager applet** *applet-name*
7. Do one of the following:
  - **event snmp oid** *oid-value* **get-type** {**exact**|**next**} **entry-op** *operator* **entry-val** *entry-value* [**exit-comb** | **and**] [**exit-op** *operator*] [**exit-val** *exit-value*] [**exit-time** *exit-time-value*] **poll-interval** *poll-int-value*
8. **action** *label* **cli command** *cli-string* [**pattern** *pattern-string*]
9. **action** *label* **syslog** [**priority** *priority-level*] **msg** *msg-text* **facility** *string*
10. **action** *label* **mail server** *server-address* **to** *to-address* **from** *from-address* [**cc** *cc-address*] **subject** *subject* **body** *body-text*
11. Add more action commands as required.
12. **end**



DETAILED STEPS

|        | Command or Action                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      | Purpose                                                                                                                                                                                                                                                                                                                          |
|--------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Step 1 | <p><b>enable</b></p> <p><b>Example:</b></p> <pre>Device&gt; enable</pre>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               | <p>Enables privileged EXEC mode.</p> <ul style="list-style-type: none"> <li>• Enter your password if prompted.</li> </ul>                                                                                                                                                                                                        |
| Step 2 | <p><b>show event manager environment</b> [<b>all</b>] <i>variable-name</i></p> <p><b>Example:</b></p> <pre>Device# show event manager environment all</pre>                                                                                                                                                                                                                                                                                                                                                                                                                                            | <p>(Optional) Displays the name and value of EEM environment variables.</p> <ul style="list-style-type: none"> <li>• The optional <b>all</b> keyword displays all the EEM environment variables.</li> <li>• The optional <i>variable-name</i> argument displays information about the specified environment variable.</li> </ul> |
| Step 3 | <p><b>configure terminal</b></p> <p><b>Example:</b></p> <pre>Device# configure terminal</pre>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          | <p>Enters global configuration mode.</p>                                                                                                                                                                                                                                                                                         |
| Step 4 | <p><b>event manager environment</b> <i>variable-name string</i></p> <p><b>Example:</b></p> <pre>Device(config)# event manager environment _email_to engineering@example.com</pre>                                                                                                                                                                                                                                                                                                                                                                                                                      | <p>Configures the value of the specified EEM environment variable.</p> <ul style="list-style-type: none"> <li>• In this example, the environment variable that holds the e-mail address to which e-mail is sent is set to <code>engineering@example.com</code>.</li> </ul>                                                       |
| Step 5 | <p>Repeat <a href="#">Alphabetical Order of EEM Action Labels</a> for all the required environment variables.</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      | <p>Repeat <a href="#">Alphabetical Order of EEM Action Labels</a> to configure all the environment variables required by the policy to be registered in <a href="#">Alphabetical Order of EEM Action Labels</a>.</p>                                                                                                             |
| Step 6 | <p><b>event manager applet</b> <i>applet-name</i></p> <p><b>Example:</b></p> <pre>Device(config)# event manager applet memory-fail</pre>                                                                                                                                                                                                                                                                                                                                                                                                                                                               | <p>Registers the applet with the Embedded Event Manager (EEM) and enters applet configuration mode.</p>                                                                                                                                                                                                                          |
| Step 7 | <p>Do one of the following:</p> <ul style="list-style-type: none"> <li>• <b>event snmp oid</b> <i>oid-value</i> <b>get-type</b> {<b>exact</b> <b>next</b>} <b>entry-op</b> <i>operator</i> <b>entry-val</b> <i>entry-value</i> [<b>exit-comb</b>   <b>and</b>] [<b>exit-op</b> <i>operator</i>] [<b>exit-val</b> <i>exit-value</i>] [<b>exit-time</b> <i>exit-time-value</i>] <b>poll-interval</b> <i>poll-int-value</i></li> </ul> <p><b>Example:</b></p> <pre>Device(config-applet)# event snmp oid 1.3.6.1.4.1.9.9.48.1.1.1.6.1 get-type exact entry-op lt entry-val 5120000 poll-interval 90</pre> | <p>Specifies the event criteria that cause the EEM applet to run.</p> <ul style="list-style-type: none"> <li>• In this example, an EEM event is triggered when free memory falls below the value of 5120000.</li> <li>• Exit criteria are optional, and if not specified, event monitoring is reenabled immediately.</li> </ul>  |

|                | Command or Action                                                                                                                                                                                                                                                                                                                                                                                                                                                                    | Purpose                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |
|----------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Step 8</b>  | <p><b>action</b> <i>label</i> <b>cli command</b> <i>cli-string</i> [<b>pattern</b> <i>pattern-string</i>]</p> <p><b>Example:</b></p> <pre>Device(config-applet)# action 1.0 cli command "enable"</pre> <p><b>Example:</b></p> <pre>Device(config-applet)# action 2.0 cli command "clear counters Ethernet0/1" pattern "confirm"</pre> <p><b>Example:</b></p> <pre>Device(config-applet)# action 3.0 cli command "y"</pre>                                                            | <p>Specifies the action of executing a Cisco IOS CLI command when an EEM applet is triggered.</p> <p>The <b>pattern</b> keyword is optional and is used only when the command string solicits input. The <b>action cli</b> command ends when the solicited prompt as specified in the optional <b>pattern</b> keyword is received. You are required to specify a regular expression pattern that will match the next solicited prompt. Specification of an incorrect pattern will cause the <b>action cli</b> command to wait forever until the applet execution times out due to the maxrun timer expiration.</p> <ul style="list-style-type: none"> <li>The action taken is to specify an EEM applet to run when the <b>pattern</b> keyword specifies the <i>confirm</i> argument for the <b>clear counters Ethernet0/1</b> command. In this case the command string solicits input, such as “confirm,” which has to be completed with a “yes” or a “no” input.</li> </ul> |
| <b>Step 9</b>  | <p><b>action</b> <i>label</i> <b>syslog</b> [<b>priority</b> <i>priority-level</i>] <b>msg</b> <i>msg-text</i> <b>facility</b> <i>string</i></p> <p><b>Example:</b></p> <pre>Device(config-applet)# action 1.0 syslog priority critical msg "Memory exhausted; current available memory is \$_snmp_oid_val bytes"</pre> <p><b>Example:</b></p> <pre>Device(config-applet)# action 1.0 syslog priority errors facility EEM-FAC message "TEST MSG"</pre>                               | <p>Specifies the action to be taken when an EEM applet is triggered.</p> <p>In this example, the action taken is to write a message to syslog.</p> <ul style="list-style-type: none"> <li>The optional <b>priority</b> keyword specifies the priority level of the syslog messages. If selected, the <i>priority-level</i> argument must be defined.</li> <li>The <i>msg-text</i> argument can be character text, an environment variable, or a combination of the two.</li> <li>The <b>facility</b> keyword specifies the location of generated message</li> <li>The <i>string</i> argument can be character text, an environment variable, or a combination of the two.</li> </ul>                                                                                                                                                                                                                                                                                         |
| <b>Step 10</b> | <p><b>action</b> <i>label</i> <b>mail server</b> <i>server-address</i> <b>to</b> <i>to-address</i> <b>from</b> <i>from-address</i> [<b>cc</b> <i>cc-address</i>] <b>subject</b> <i>subject</i> <b>body</b> <i>body-text</i></p> <p><b>Example:</b></p> <pre>Device(config-applet)# action 2.0 mail server 192.168.1.10 to engineering@example.com from devtest@example.com subject "Memory failure" body "Memory exhausted; current available memory is \$_snmp_oid_val bytes"</pre> | <p>Specifies the action of sending a short e-mail when an EEM applet is triggered.</p> <ul style="list-style-type: none"> <li>The <i>server-address</i> argument specifies the fully qualified domain name of the e-mail server to be used to forward the e-mail.</li> <li>The <i>to-address</i> argument specifies the e-mail address where the e-mail is to be sent.</li> <li>The <i>from-address</i> argument specifies the e-mail address from which the e-mail is sent.</li> <li>The <i>subject</i> argument specifies the subject line content of the e-mail as an alphanumeric string.</li> </ul>                                                                                                                                                                                                                                                                                                                                                                     |

|                | Command or Action                                                      | Purpose                                                                                                                                             |
|----------------|------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------|
|                |                                                                        | <ul style="list-style-type: none"> <li>The <i>body-text</i> argument specifies the text content of the e-mail as an alphanumeric string.</li> </ul> |
| <b>Step 11</b> | Add more action commands as required.                                  | --                                                                                                                                                  |
| <b>Step 12</b> | <b>end</b><br><b>Example:</b><br><pre>Device(config-applet)# end</pre> | Exits applet configuration mode and returns to privileged EXEC mode.                                                                                |

## Troubleshooting Tips

Use the **debug event manager** command in privileged EXEC mode to troubleshoot EEM command operations. Use any debugging command with caution as the volume of generated output can slow or stop the device operations. We recommend that this command be used only under the supervision of a Cisco engineer.

## Registering and Defining an EEM Tcl Script

Perform this task to configure environment variables and register an EEM policy. EEM schedules and runs policies on the basis of an event specification that is contained within the policy itself. When an EEM policy is registered, the software examines the policy and registers it to be run when the specified event occurs.

### Before you begin

You must have a policy available that is written in the Tcl scripting language. Sample policies are provided--see the details in the [Sample EEM Policies, on page 2093](#) to see which policies are available for the Cisco IOS release image that you are using--and these sample policies are stored in the system policy directory.

### SUMMARY STEPS

- enable**
- show event manager environment** [**all**| *variable-name*]
- configure terminal**
- event manager environment** *variable-name string*
- Repeat [Registering and Defining an EEM Tcl Script](#) to configure all the environment variables required by the policy to be registered in [Registering and Defining an EEM Tcl Script](#).
- event manager policy** *policy-filename* [**type** {**system**| **user**}] [**trap**]
- exit**

### DETAILED STEPS

|               | Command or Action                                                | Purpose                                                                                                          |
|---------------|------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------|
| <b>Step 1</b> | <b>enable</b><br><b>Example:</b><br><pre>Device&gt; enable</pre> | Enables privileged EXEC mode. <ul style="list-style-type: none"> <li>Enter your password if prompted.</li> </ul> |

|               | Command or Action                                                                                                                                                                                                    | Purpose                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
|---------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Step 2</b> | <p><b>show event manager environment</b> [<b>all</b>  <i>variable-name</i>]</p> <p><b>Example:</b></p> <pre>Device# show event manager environment all</pre>                                                         | <p>(Optional) Displays the name and value of EEM environment variables.</p> <ul style="list-style-type: none"> <li>• The optional <b>all</b> keyword displays all the EEM environment variables.</li> <li>• The optional <i>variable-name</i> argument displays information about the specified environment variable.</li> </ul>                                                                                                                                                                                                       |
| <b>Step 3</b> | <p><b>configure terminal</b></p> <p><b>Example:</b></p> <pre>Device# configure terminal</pre>                                                                                                                        | Enters global configuration mode.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |
| <b>Step 4</b> | <p><b>event manager environment</b> <i>variable-name string</i></p> <p><b>Example:</b></p> <pre>Device(config)# event manager environment _cron_entry 0-59/2 0-23/1 * * 0-6</pre>                                    | <p>Configures the value of the specified EEM environment variable.</p> <ul style="list-style-type: none"> <li>• In this example, the software assigns a CRON timer environment variable to be set to the second minute of every hour of every day.</li> </ul>                                                                                                                                                                                                                                                                          |
| <b>Step 5</b> | Repeat <a href="#">Registering and Defining an EEM Tcl Script</a> to configure all the environment variables required by the policy to be registered in <a href="#">Registering and Defining an EEM Tcl Script</a> . | --                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |
| <b>Step 6</b> | <p><b>event manager policy</b> <i>policy-filename</i> [<b>type</b> {<b>system</b> <b>user</b>}] [<b>trap</b>]</p> <p><b>Example:</b></p> <pre>Device(config)# event manager policy tm_cli_cmd.tcl type system</pre>  | <p>Registers the EEM policy to be run when the specified event defined within the policy occurs.</p> <ul style="list-style-type: none"> <li>• Use the <b>system</b> keyword to register a Cisco-defined system policy.</li> <li>• Use the <b>user</b> keyword to register a user-defined system policy.</li> <li>• Use the <b>trap</b> keyword to generate an SNMP trap when the policy is triggered.</li> <li>• In this example, the sample EEM policy named <code>tm_cli_cmd.tcl</code> is registered as a system policy.</li> </ul> |
| <b>Step 7</b> | <p><b>exit</b></p> <p><b>Example:</b></p> <pre>Device(config)# exit</pre>                                                                                                                                            | Exits global configuration mode and returns to privileged EXEC mode.                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |

### Examples

In the following example, the **show event manager environment** privileged EXEC command is used to display the name and value of all EEM environment variables.

```

Device# show event manager environment all
No. Name Value
1 _cron_entry 0-59/2 0-23/1 * * 0-6
2 _show_cmd show ver
3 _syslog_pattern .*UPDOWN.*Ethernet1/0.*
4 _config_cmd1 interface Ethernet1/0
5 _config_cmd2 no shut

```

## Unregistering Embedded Event Manager Policies

Perform this task to remove an EEM policy from the running configuration file. Execution of the policy is canceled.

### SUMMARY STEPS

1. **enable**
2. **show event manager policy registered** [description [*policy-name*] | detailed *policy-filename* [system | user] | [event-type *event-name*] [system | user] [time-ordered | name-ordered]]
3. **configure terminal**
4. **no event manager policy** *policy-filename*
5. **exit**
6. Repeat Step 2 to ensure that the policy has been removed.

### DETAILED STEPS

|        | Command or Action                                                                                                                                                                                                                                                                     | Purpose                                                                                                                                                                                                                                                                                                                                        |
|--------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Step 1 | <b>enable</b><br><b>Example:</b><br>Device> enable                                                                                                                                                                                                                                    | Enables privileged EXEC mode. <ul style="list-style-type: none"> <li>• Enter your password if prompted.</li> </ul>                                                                                                                                                                                                                             |
| Step 2 | <b>show event manager policy registered</b> [description [ <i>policy-name</i> ]   detailed <i>policy-filename</i> [system   user]   [event-type <i>event-name</i> ] [system   user] [time-ordered   name-ordered]]<br><b>Example:</b><br>Device# show event manager policy registered | (Optional) Displays the EEM policies that are currently registered. <ul style="list-style-type: none"> <li>• The optional <b>system</b> and <b>user</b> keywords display the registered system and user policies.</li> <li>• If no keywords are specified, EEM registered policies for all event types are displayed in time order.</li> </ul> |
| Step 3 | <b>configure terminal</b><br><b>Example:</b><br>Device# configure terminal                                                                                                                                                                                                            | Enters global configuration mode.                                                                                                                                                                                                                                                                                                              |
| Step 4 | <b>no event manager policy</b> <i>policy-filename</i><br><b>Example:</b><br>Device(config)# no event manager policy IPSLAping1                                                                                                                                                        | Removes the EEM policy from the configuration, causing the policy to be unregistered.                                                                                                                                                                                                                                                          |

|               | Command or Action                                                                                                            | Purpose                                                              |
|---------------|------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------|
| <b>Step 5</b> | <b>exit</b><br><b>Example:</b><br>Device(config)# exit                                                                       | Exits global configuration mode and returns to privileged EXEC mode. |
| <b>Step 6</b> | Repeat Step 2 to ensure that the policy has been removed.<br><b>Example:</b><br>Device# show event manager policy registered | --                                                                   |

### Examples

In the following example, the **show event manager policy registered** privileged EXEC command is used to display the two EEM applets that are currently registered:

```
Device# show event manager policy registered
No. Class Type Event Type Trap Time Registered Name
1 applet system snmp Off Fri Aug 12 17:42:52 2005 IPSLAping1
oid {1.3.6.1.4.1.9.9.42.1.2.9.1.6.4} get-type exact entry-op eq entry-val {1}
exit-op eq exit-val {2} poll-interval 90.000
action 1.0 syslog priority critical msg "Server IPEcho Failed: OID=$_snmp_oid_val"
action 1.1 snmp-trap strdata "EEM detected server reachability failure to 10.1.88.9"
action 1.2 publish-event sub-system 88000101 type 1 arg1 "10.1.88.9" arg2 "IPSLAEcho"
arg3 "fail"
action 1.3 counter name _IPSLA1F op inc value 1
2 applet system snmp Off Thu Sep 15 05:57:16 2005 memory-fail
oid {1.3.6.1.4.1.9.9.48.1.1.1.6.1} get-type exact entry-op lt entry-val {5120000}
poll-interval 90
action 1.0 syslog priority critical msg Memory exhausted; current available memory is
$_snmp_oid_val bytes
action 2.0 force-switchover
```

In the following example, the **show event manager policy registered** privileged EXEC command is used to show that applet IPSLAping1 has been removed after entering the **no event manager policy** command:

```
Device# show event manager policy registered
No. Class Type Event Type Trap Time Registered Name
1 applet system snmp Off Thu Sep 15 05:57:16 2005 memory-fail
oid {1.3.6.1.4.1.9.9.48.1.1.1.6.1} get-type exact entry-op lt entry-val {5120000}
poll-interval 90
action 1.0 syslog priority critical msg Memory exhausted; current available memory is
$_snmp_oid_val bytes
action 2.0 force-switchover
```

## Suspending All Embedded Event Manager Policy Execution

Perform this task to immediately suspend the execution of all EEM policies. Suspending policies, instead of unregistering them might be necessary for reasons of temporary performance or security.

## SUMMARY STEPS

1. **enable**
2. **show event manager policy registered** [description [*policy-name*] | **detailed** *policy-filename* [system | user] | [event-type *event-name*] [system | user] [time-ordered | name-ordered]]
3. **configure terminal**
4. **event manager scheduler suspend**
5. **exit**

## DETAILED STEPS

|        | Command or Action                                                                                                                                                                                                                                                                            | Purpose                                                                                                                                                                                                                                                                                                                                        |
|--------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Step 1 | <b>enable</b><br><b>Example:</b><br>Device> enable                                                                                                                                                                                                                                           | Enables privileged EXEC mode. <ul style="list-style-type: none"> <li>• Enter your password if prompted.</li> </ul>                                                                                                                                                                                                                             |
| Step 2 | <b>show event manager policy registered</b> [description [ <i>policy-name</i> ]   <b>detailed</b> <i>policy-filename</i> [system   user]   [event-type <i>event-name</i> ] [system   user] [time-ordered   name-ordered]]<br><b>Example:</b><br>Device# show event manager policy registered | (Optional) Displays the EEM policies that are currently registered. <ul style="list-style-type: none"> <li>• The optional <b>system</b> and <b>user</b> keywords display the registered system and user policies.</li> <li>• If no keywords are specified, EEM registered policies for all event types are displayed in time order.</li> </ul> |
| Step 3 | <b>configure terminal</b><br><b>Example:</b><br>Device# configure terminal                                                                                                                                                                                                                   | Enters global configuration mode.                                                                                                                                                                                                                                                                                                              |
| Step 4 | <b>event manager scheduler suspend</b><br><b>Example:</b><br>Device(config)# event manager scheduler suspend                                                                                                                                                                                 | Immediately suspends the execution of all EEM policies.                                                                                                                                                                                                                                                                                        |
| Step 5 | <b>exit</b><br><b>Example:</b><br>Device(config)# exit                                                                                                                                                                                                                                       | Exits global configuration mode and returns to privileged EXEC mode.                                                                                                                                                                                                                                                                           |

## Displaying Embedded Event Manager History Data

Perform this optional task to change the size of the history tables and to display EEM history data.

## SUMMARY STEPS

1. **enable**
2. **configure terminal**

3. **event manager history size** {events | traps} [size]
4. **exit**
5. **show event manager history events** [detailed] [maximum number]
6. **show event manager history traps** {server | policy}

## DETAILED STEPS

---

### Step 1 enable

Enables privileged EXEC mode. Enter your password if prompted.

#### Example:

```
Device> enable
```

### Step 2 configure terminal

Enters global configuration mode.

#### Example:

```
Device# configure terminal
```

### Step 3 event manager history size {events | traps} [size]

Use this command to change the size of the EEM event history table or the size of the EEM SNMP trap history table. In the following example, the size of the EEM event history table is changed to 30 entries:

#### Example:

```
Device(config)# event manager history size events 30
```

### Step 4 exit

Exits global configuration mode and returns to privileged EXEC mode.

#### Example:

```
Device(config)# exit
```

### Step 5 show event manager history events [detailed] [maximum number]

Use this command to display detailed information about each EEM event, for example:

#### Example:

```
Device# show event manager history events
No. Time of Event Event Type Name
1 Fri Aug13 21:42:57 2004 snmp applet: SAAping1
2 Fri Aug13 22:20:29 2004 snmp applet: SAAping1
3 Wed Aug18 21:54:48 2004 snmp applet: SAAping1
4 Wed Aug18 22:06:38 2004 snmp applet: SAAping1
5 Wed Aug18 22:30:58 2004 snmp applet: SAAping1
6 Wed Aug18 22:34:58 2004 snmp applet: SAAping1
7 Wed Aug18 22:51:18 2004 snmp applet: SAAping1
8 Wed Aug18 22:51:18 2004 application applet: CustApp1
```

### Step 6 show event manager history traps {server | policy}



Use this command to display the EEM SNMP traps that have been sent either from the EEM server or from an EEM policy. In the following example, the EEM SNMP traps that were triggered from within an EEM policy are displayed.

**Example:**

```
Device# show event manager history traps policy
No. Time Trap Type Name
1 Wed Aug18 22:30:58 2004 policy EEM Policy Director
2 Wed Aug18 22:34:58 2004 policy EEM Policy Director
3 Wed Aug18 22:51:18 2004 policy EEM Policy Director
```

## Displaying Embedded Event Manager Registered Policies

Perform this optional task to display registered EEM policies.

### SUMMARY STEPS

1. **enable**
2. **show event manager policy registered [event-type *event-name*] [time-ordered| name-ordered]**

### DETAILED STEPS

#### Step 1 **enable**

Enables privileged EXEC mode. Enter your password if prompted.

**Example:**

```
Device> enable
```

#### Step 2 **show event manager policy registered [event-type *event-name*] [time-ordered| name-ordered]**

Use this command with the **time-ordered** keyword to display information about currently registered policies sorted by time, for example:

**Example:**

```
Device# show event manager policy registered time-ordered
No. Type Event Type Time Registered Name
1 applet snmp Thu May30 05:57:16 2004 memory-fail
 oid {1.3.6.1.4.1.9.9.48.1.1.1.6.1} get-type exact entry-op lt entry-val
 {5120000} poll-interval 90
 action 1.0 syslog priority critical msg "Memory exhausted; current available memory
 is $_snmp_oid_val bytes"
 action 2.0 force-switchover
2 applet syslog Wed Jul16 00:05:17 2004 intf-down
 pattern {.*UPDOWN.*Ethernet1/0.*}
 action 1.0 cns-event msg "Interface state change: $_syslog_msg"
```

Use this command with the **name-ordered** keyword to display information about currently registered policies sorted by name, for example:

**Example:**

```
Device# show event manager policy registered name-ordered
No. Type Event Type Time Registered Name
1 applet syslog Wed Jul16 00:05:17 2004 intf-down
 pattern {.*UPDOWN.*Ethernet1/0.*}
 action 1.0 cns-event msg "Interface state change: $_syslog_msg"
2 applet snmp Thu May30 05:57:16 2004 memory-fail
 oid {1.3.6.1.4.1.9.9.48.1.1.1.6.1} get-type exact entry-op lt entry-val
 {5120000} poll-interval 90
 action 1.0 syslog priority critical msg "Memory exhausted; current available memory
is $_snmp_oid_val bytes"
 action 2.0 force-switchover
```

Use this command with the **event-type** keyword to display information about currently registered policies for the event type specified in the *event-name* argument, for example:

#### Example:

```
Device# show event manager policy registered event-type syslog
No. Type Event Type Time Registered Name
1 applet syslog Wed Jul16 00:05:17 2004 intf-down
 pattern {.*UPDOWN.*Ethernet1/0.*}
 action 1.0 cns-event msg "Interface state change: $_syslog_msg"
```

## Configuring Event SNMP Notification

Perform this task to configure SNMP notifications.

#### Before you begin

- SNMP event manager must be configured using the **snmp-server manager** command.

#### SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **event manager applet** *applet-name*
4. **event** [**tag** *event-tag*] **snmp-notification oid** *oid-string* **oid-val** *comparison-value* **op** *operator* [**maxrun** *maxruntime-number*] [**src-ip-address** *ip-address*] [**dest-ip-address** *ip-address*] [**default** *seconds*] [**direction** {**incoming** | **outgoing**}] [**msg-op** {**drop** | **send**}]
5. **end**

#### DETAILED STEPS

|        | Command or Action                                          | Purpose                                                                                                            |
|--------|------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------|
| Step 1 | <b>enable</b><br><br><b>Example:</b><br><br>Device> enable | Enables privileged EXEC mode. <ul style="list-style-type: none"> <li>• Enter your password if prompted.</li> </ul> |

|        | Command or Action                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              | Purpose                                                                                                                                                 |
|--------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------|
| Step 2 | <b>configure terminal</b><br><b>Example:</b><br><pre>Device# configure terminal</pre>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          | Enters global configuration mode.                                                                                                                       |
| Step 3 | <b>event manager applet</b> <i>applet-name</i><br><b>Example:</b><br><pre>Device(config)# event manager applet snmp</pre>                                                                                                                                                                                                                                                                                                                                                                                                                                                      | Registers the applet with the event manager server and enters applet configuration mode.                                                                |
| Step 4 | <b>event</b> [ <b>tag</b> <i>event-tag</i> ] <b>snmp-notification oid</b> <i>oid-string</i> <b>oid-val</b> <i>comparison-value</i> <b>op</b> <i>operator</i> [ <b>maxrun</b> <i>maxruntime-number</i> ] [ <b>src-ip-address</b> <i>ip-address</i> ] [ <b>dest-ip-address</b> <i>ip-address</i> ] [ <b>default</b> <i>seconds</i> ] [ <b>direction</b> { <b>incoming</b>   <b>outgoing</b> }] [ <b>msg-op</b> { <b>drop</b>   <b>send</b> }]<br><b>Example:</b><br><pre>Device(config-applet)# event snmp-notification dest-ip-address 192.168.1.1 oid 1 op eq oid-val 10</pre> | Specifies the event criteria for an Embedded Event Manager (EEM) applet that is run by sampling Simple Network Management Protocol (SNMP) notification. |
| Step 5 | <b>end</b><br><b>Example:</b><br><pre>Device(config-applet)# end</pre>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         | Exits applet configuration mode and returns to privileged EXEC mode.                                                                                    |

## Configuring Multiple Event Support

The multiple event support feature adds the ability to register multiple events in the EEM server. The multiple event support involves one or more event occurrences, one or more tracked object states, and a time period for the event to occur. The event parameters are specified in the CLI commands. The data structure to handle multiple events contains multiple event identifiers and correlation logic. This data is used to register multiple events in the EEM Server.

### Setting the Event Configuration Parameters

The **trigger** command enters the trigger applet configuration mode and specifies the multiple event configuration statements for EEM applets. The trigger statement is used to relate multiple event statement using the *tag* argument specified in each event statement. The events are raised based on the specified parameters.

#### SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **event manager applet** *applet-name*
4. **event** [**tag** *event-tag*] **cli pattern** *regular-expression* **sync** {**yes** | **no skip** {**yes** | **no**}} [**occurs** *num-occurrences*] [**period** *period-value*] [**maxrun** *maxruntime-number*]
5. **trigger** [**occurs** *occurs-value*] [**period** *period-value*] [**period-start** *period-start-value*] [**delay** *delay-value*]

6. **correlate** {**event** *event-tag* | **track** *object-number*} [*boolean-operator* **event** *event-tag*]
7. **attribute** **tag** *event-tag* [**occurs** *occurs-value*]
8. **action** *label* **cli command** *cli-string*

## DETAILED STEPS

|               | Command or Action                                                                                                                                                                                                                                                                                                                                                                                                      | Purpose                                                                                                                                                                                                                           |
|---------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Step 1</b> | <b>enable</b><br><b>Example:</b><br>Device> enable                                                                                                                                                                                                                                                                                                                                                                     | Enables privileged EXEC mode. <ul style="list-style-type: none"> <li>• Enter your password if prompted.</li> </ul>                                                                                                                |
| <b>Step 2</b> | <b>configure terminal</b><br><b>Example:</b><br>Device# configure terminal                                                                                                                                                                                                                                                                                                                                             | Enters global configuration mode.                                                                                                                                                                                                 |
| <b>Step 3</b> | <b>event manager applet</b> <i>applet-name</i><br><b>Example:</b><br>Device(config)# event manager applet EventInterface                                                                                                                                                                                                                                                                                               | Registers an applet with EEM and enters applet configuration mode.                                                                                                                                                                |
| <b>Step 4</b> | <b>event</b> [ <b>tag</b> <i>event-tag</i> ] <b>cli pattern</b> <i>regular-expression</i> <b>sync</b> { <b>yes</b>   <b>no skip</b> { <b>yes</b>   <b>no</b> }} [ <b>occurs</b> <i>num-occurrences</i> ] [ <b>period</b> <i>period-value</i> ] [ <b>maxrun</b> <i>maxruntime-number</i> ]<br><b>Example:</b><br>Device(config-applet)# event tag 1.0 cli pattern "show bgp all" sync yes occurs 32 period 60 maxrun 60 | Specifies the event criteria for an EEM applet that is run by matching a Cisco IOS command-line interface (CLI) command.                                                                                                          |
| <b>Step 5</b> | <b>trigger</b> [ <b>occurs</b> <i>occurs-value</i> ] [ <b>period</b> <i>period-value</i> ] [ <b>period-start</b> <i>period-start-value</i> ] [ <b>delay</b> <i>delay-value</i> ]<br><b>Example:</b><br>Device(config-applet)# trigger occurs 1 period-start "0 8 * * 1-5" period 60                                                                                                                                    | Specifies the complex event configuration parameters for an EEM applet.                                                                                                                                                           |
| <b>Step 6</b> | <b>correlate</b> { <b>event</b> <i>event-tag</i>   <b>track</b> <i>object-number</i> } [ <i>boolean-operator</i> <b>event</b> <i>event-tag</i> ]<br><b>Example:</b><br>Device(config-applet)# correlate event 1.0 or event 2.0                                                                                                                                                                                         | Specifies a complex event correlation in the trigger mode for an EEM applet.<br><br><b>Note</b> When "and" is used to group events such as traps or syslog messages, then the default trigger occurrence window is three minutes. |
| <b>Step 7</b> | <b>attribute</b> <b>tag</b> <i>event-tag</i> [ <b>occurs</b> <i>occurs-value</i> ]<br><b>Example:</b><br>Device(config-applet)# attribute tag 1.0 occurs 1                                                                                                                                                                                                                                                             | Specifies up to eight attribute statements to build a complex event for an EEM applet.                                                                                                                                            |

|        | Command or Action                                                                                                                                  | Purpose                                                                          |
|--------|----------------------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------|
| Step 8 | <b>action</b> <i>label</i> <b>cli command</b> <i>cli-string</i><br><b>Example:</b><br>Device(config-applet)# action 1.0 cli command "show pattern" | Specifies the action of executing a CLI command when an EEM applet is triggered. |

### Examples

In the following example, applet is run if the **show bgp all** CLI command and any syslog message that contains the string "COUNT" occurred within a period 60 seconds.

```

event manager applet delay_50
 event tag 1.0 cli pattern "show bgp all" sync yes occurs 32 period 60 maxrun 60
 event tag 2.0 syslog pattern "COUNT"
 trigger occurs 1 delay 50
 correlate event 1.0 or event 2.0
 attribute tag 1.0 occurs 1
 attribute tag 2.0 occurs 1
 action 1.0 cli command "show pattern"
 action 2.0 cli command "enable"
 action 3.0 cli command "config terminal"
 action 4.0 cli command " ip route 192.0.2.0 255.255.255.224 192.0.2.12"
 action 91.0 cli command "exit"
 action 99.0 cli command "show ip route | incl 192.0.2.5"

```

## Configuring EEM Class-Based Scheduling

To schedule Embedded Event Manager (EEM) policies and set policy scheduling options, perform this task. In this task, two EEM execution threads are created to run applets assigned to the default class.

The EEM policies will be assigned a class using the **class** keyword when they are registered. EEM policies registered without a class will be assigned to the default class. Threads that have default class, will service the default class when the thread is available for work. Threads that are assigned specific class letters will service any policy with a matching class letter when the thread is available for work.

If there is no EEM execution thread available to run the policy in the specified class and a scheduler rule for the class is configured, the policy will wait until a thread of that class is available for execution. Synchronous policies that are triggered from the same input event should be scheduled in the same execution thread.

### SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **event manager scheduler** {*applet* | *axp* | *call-home*} **thread class** *class-options* **number** *thread-number*
4. **exit**

## DETAILED STEPS

|               | Command or Action                                                                                                                                                                                                         | Purpose                                                                                                                                                                                                             |
|---------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Step 1</b> | <b>enable</b><br><b>Example:</b><br>Device> enable                                                                                                                                                                        | Enables privileged EXEC mode. <ul style="list-style-type: none"> <li>• Enter your password if prompted.</li> </ul>                                                                                                  |
| <b>Step 2</b> | <b>configure terminal</b><br><b>Example:</b><br>Device# configure terminal                                                                                                                                                | Enters global configuration mode.                                                                                                                                                                                   |
| <b>Step 3</b> | <b>event manager scheduler {applet   axp   call-home}</b><br><b>thread class class-options number thread-number</b><br><b>Example:</b><br>Device(config)# event manager scheduler applet<br>thread class default number 2 | Schedules EEM policies and sets policy scheduling options. <ul style="list-style-type: none"> <li>• In this example, two EEM execution threads are created to run applets assigned to the default class.</li> </ul> |
| <b>Step 4</b> | <b>exit</b><br><b>Example:</b><br>Device(config)# exit                                                                                                                                                                    | Exits global configuration mode and returns to privileged EXEC mode.                                                                                                                                                |

## Holding a Scheduled EEM Policy Event or Event Queue

To hold a scheduled EEM policy event or event queue in the EEM scheduler, perform this task. In this task, all pending EEM policies are displayed. A policy identified using a job ID of 2 is held in the EEM scheduler, and the final step shows that the policy with a job ID of 2 has changed status from pending to held.

## SUMMARY STEPS

1. **enable**
2. **show event manager policy pending [queue-type {applet | call-home | axp | script} class class-options | detailed]**
3. **event manager scheduler hold {all | policy job-id | queue-type {applet | call-home | axp | script} class class-options} [processor {rp\_primary| rp\_standby}]**
4. **show event manager policy pending [queue-type {applet | call-home | axp | script} class class-options | detailed]**

## DETAILED STEPS

|               | Command or Action                                  | Purpose                                                                                                            |
|---------------|----------------------------------------------------|--------------------------------------------------------------------------------------------------------------------|
| <b>Step 1</b> | <b>enable</b><br><b>Example:</b><br>Device> enable | Enables privileged EXEC mode. <ul style="list-style-type: none"> <li>• Enter your password if prompted.</li> </ul> |

|        | Command or Action                                                                                                                                                                                                                                           | Purpose                                                                                                                                                                                       |
|--------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Step 2 | <p><b>show event manager policy pending</b> [queue-type {applet   call-home   axp   script} class class-options   detailed]</p> <p><b>Example:</b></p> <pre>Device# show event manager policy pending</pre>                                                 | Displays the pending EEM policies.                                                                                                                                                            |
| Step 3 | <p><b>event manager scheduler hold</b> {all   policy job-id   queue-type {applet   call-home   axp   script} class class-options} [processor {rp_primary   rp_standby}]</p> <p><b>Example:</b></p> <pre>Device# event manager scheduler hold policy 2</pre> | <p>Holds a scheduled EEM policy event or event queue in the EEM scheduler.</p> <ul style="list-style-type: none"> <li>In this example, a policy with a job ID of 2 is put on hold.</li> </ul> |
| Step 4 | <p><b>show event manager policy pending</b> [queue-type {applet   call-home   axp   script} class class-options   detailed]</p> <p><b>Example:</b></p> <pre>Device# show event manager policy pending</pre>                                                 | Displays the status of EEM policy put on hold in Step 3 as held, along with other pending policies.                                                                                           |

### Examples

The following example shows how to view all pending EEM policies and to hold the EEM policy with a job ID of 2.

```
Device# show event manager policy pending
no. job id status time of event event type name
1 1 pend Thu Sep 7 02:54:04 2006 syslog applet: one
2 2 pend Thu Sep 7 02:54:04 2006 syslog applet: two
3 3 pend Thu Sep 7 02:54:04 2006 syslog applet: three
Device# event manager scheduler hold policy 2
Device# show event manager policy pending

no. job id status time of event event type name
1 1 pend Thu Sep 7 02:54:04 2006 syslog applet: one
2 2 held Thu Sep 7 02:54:04 2006 syslog applet: two
3 3 pend Thu Sep 7 02:54:04 2006 syslog applet: three
```

## Resuming Execution of EEM Policy Events or Event Queues

To resume the execution of specified EEM policies, perform this task. In this task, the policy that was put on hold in the Holding a Scheduled EEM Policy Event or Event Queue task is now allowed to resume execution.

### SUMMARY STEPS

- enable
- show event manager policy pending
- event manager scheduler release {all | policy policy-id | queue-type {applet | call-home | axp | script}} class class-options [processor {rp\_primary | rp\_standby}]
- show event manager policy pending

## DETAILED STEPS

|               | Command or Action                                                                                                                                                                                                                                                          | Purpose                                                                                                                                                                                                                                                          |
|---------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Step 1</b> | <b>enable</b><br><b>Example:</b><br><pre>Device&gt; enable</pre>                                                                                                                                                                                                           | Enables privileged EXEC mode. <ul style="list-style-type: none"> <li>• Enter your password if prompted.</li> </ul>                                                                                                                                               |
| <b>Step 2</b> | <b>show event manager policy pending</b><br><b>Example:</b><br><pre>Device# show event manager policy pending</pre>                                                                                                                                                        | Displays the pending and held EEM policies.<br><b>Note</b> Only the syntax applicable to this task is used in this example. For more details, see the Cisco IOS Network Management Command Reference.                                                            |
| <b>Step 3</b> | <b>event manager scheduler release</b> {all   policy <i>policy-id</i>   queue-type {applet   call-home   axp   script}} class <i>class-options</i> [processor {rp_primary   rp_standby}]<br><b>Example:</b><br><pre>Device# event manager scheduler release policy 2</pre> | Resumes execution of specified EEM policies. <ul style="list-style-type: none"> <li>• The example shows how to resume the execution of the policy with job ID of 2.</li> </ul>                                                                                   |
| <b>Step 4</b> | <b>show event manager policy pending</b><br><b>Example:</b><br><pre>Device# show event manager policy pending</pre>                                                                                                                                                        | Displays the status of the EEM policy resumed in Step 3 as pending, along with other pending policies.<br><b>Note</b> Only the syntax applicable to this task is used in this example. For more details, see the Cisco IOS Network Management Command Reference. |

## Examples

The following example shows how to view all pending EEM policies, to specify the policy that will resume execution, and to see that the policy is now back in a pending status.

```
Device# show event manager policy pending

no. job id status time of event event type name
1 1 pend Thu Sep 7 02:54:04 2006 syslog applet: one
2 2 held Thu Sep 7 02:54:04 2006 syslog applet: two
3 3 pend Thu Sep 7 02:54:04 2006 syslog applet: three
Rotuer# event manager scheduler release policy 2
Rotuer# show event manager policy pending

no. job id status time of event event type name
1 1 pend Thu Sep 7 02:54:04 2006 syslog applet: one
2 2 pend Thu Sep 7 02:54:04 2006 syslog applet: two
3 3 pend Thu Sep 7 02:54:04 2006 syslog applet: three
```

## Clearing Pending EEM Policy Events or Event Queues

Perform this task to clear EEM policies that are executing or pending execution. In this task, the EEM policy with a job ID of 2 is cleared from the pending queue. The **show event manager policy pending** command is used to display the policies that are pending before and after the policy is cleared.



## SUMMARY STEPS

1. `enable`
2. `show event manager policy pending`
3. `event manager scheduler clear {all | policy job-id | queue-type {applet | call-home | axp | script} class class-options} [processor {rp_primary | rp_standby}]`
4. `show event manager policy pending`

## DETAILED STEPS

|        | Command or Action                                                                                                                                                                                                                                                                 | Purpose                                                                                                                                                                                                                                         |
|--------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Step 1 | <p><code>enable</code></p> <p><b>Example:</b></p> <pre>Device&gt; enable</pre>                                                                                                                                                                                                    | <p>Enables privileged EXEC mode.</p> <ul style="list-style-type: none"> <li>• Enter your password if prompted.</li> </ul>                                                                                                                       |
| Step 2 | <p><code>show event manager policy pending</code></p> <p><b>Example:</b></p> <pre>Device# show event manager policy pending</pre>                                                                                                                                                 | <p>Displays the pending EEM policies.</p> <p><b>Note</b> Only the syntax applicable to this task is used in this example. For more details, see the Cisco IOS Network Management Command Reference.</p>                                         |
| Step 3 | <p><code>event manager scheduler clear {all   policy <i>job-id</i>   queue-type {applet   call-home   axp   script} class <i>class-options</i>} [processor {rp_primary   rp_standby}]</code></p> <p><b>Example:</b></p> <pre>Device# event manager scheduler clear policy 2</pre> | <p>Clears EEM policies that are executing or pending execution.</p> <ul style="list-style-type: none"> <li>• In this example, the EEM policy with a job ID of 2 is cleared from the pending queue.</li> </ul>                                   |
| Step 4 | <p><code>show event manager policy pending</code></p> <p><b>Example:</b></p> <pre>Device# show event manager policy pending</pre>                                                                                                                                                 | <p>Displays all the pending EEM policies except the policy cleared in Step 3.</p> <p><b>Note</b> Only the syntax applicable to this task is used in this example. For more details, see the Cisco IOS Network Management Command Reference.</p> |

## Examples

The following example shows how to clear the EEM policy with a job ID of 2 that was pending execution. The **show** commands are used to display the policies that are pending before and after the policy is cleared.

```
Device# show event manager policy pending
no. job id status time of event event type name
1 1 pend Thu Sep 7 02:54:04 2006 syslog applet: one
2 2 pend Thu Sep 7 02:54:04 2006 syslog applet: two
3 3 pend Thu Sep 7 02:54:04 2006 syslog applet: three

Device# event manager scheduler clear policy 2
Device# show event manager policy pending
```

```

no. job id status time of event event type name
1 1 pend Thu Sep 7 02:54:04 2006 syslog applet: one
3 3 pend Thu Sep 7 02:54:04 2006 syslog applet: three

```

## Modifying the Scheduling Parameters of EEM Policy Events or Event Queues

To modify the scheduling parameters of the EEM policies, perform this task. The **show event manager policy pending** command displays policies that are assigned to the B or default class. All the currently pending policies are then changed to class A. After the configuration modification, the **show event manager policy pending** command shows all policies assigned as class A.

### SUMMARY STEPS

1. **enable**
2. **show event manager policy pending**
3. **event manager scheduler modify** {all | policy *job-id* | queue-type {applet | call-home | axp | script} | class *class-options*} [queue-priority {high | last | low | normal}][processor {rp\_primary | rp\_standby}]
4. **show event manager policy pending**

### DETAILED STEPS

|        | Command or Action                                                                                                                                                                                                                                                                                             | Purpose                                                                                                                                                                                                                                   |
|--------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Step 1 | <b>enable</b><br><b>Example:</b><br>Device> enable                                                                                                                                                                                                                                                            | Enables privileged EXEC mode. <ul style="list-style-type: none"> <li>• Enter your password if prompted.</li> </ul>                                                                                                                        |
| Step 2 | <b>show event manager policy pending</b><br><b>Example:</b><br>Device# show event manager policy pending                                                                                                                                                                                                      | Displays the pending EEM policies.<br><b>Note</b> Only the syntax applicable to this task is used in this example. For more details, see the Cisco IOS Network Management Command Reference.                                              |
| Step 3 | <b>event manager scheduler modify</b> {all   policy <i>job-id</i>   queue-type {applet   call-home   axp   script}   class <i>class-options</i> } [queue-priority {high   last   low   normal}][processor {rp_primary   rp_standby}]<br><b>Example:</b><br>Device# event manager scheduler modify all class A | Modifies the scheduling parameters of the EEM policies. <ul style="list-style-type: none"> <li>• In this example, all currently pending EEM policies are assigned to class A.</li> </ul>                                                  |
| Step 4 | <b>show event manager policy pending</b><br><b>Example:</b><br>Device# show event manager policy pending                                                                                                                                                                                                      | Displays the EEM policies modified in Step 3 along with other pending policies.<br><b>Note</b> Only the syntax applicable to this task is used in this example. For more details, see the Cisco IOS Network Management Command Reference. |

## Examples

The following example shows how to modify the scheduling parameters of the EEM policies. In this example, the **show event manager policy pending** command displays policies that are assigned to the B or default class. All the currently pending policies are then changed to class A. After the configuration modification, the **show event manager policy pending** command verifies that all policies are now assigned as class A.

```
Device# show event manager policy pending
no. class status time of event event type name
1 default pend Thu Sep 7 02:54:04 2006 syslog applet: one
2 default pend Thu Sep 7 02:54:04 2006 syslog applet: two
3 B pend Thu Sep 7 02:54:04 2006 syslog applet: three

Device# event manager scheduler modify all class A
Device# show event manager policy pending

no. class status time of event event type name
1 A pend Thu Sep 7 02:54:04 2006 syslog applet: one
2 A pend Thu Sep 7 02:54:04 2006 syslog applet: two
3 A pend Thu Sep 7 02:54:04 2006 syslog applet: three
```

## Verifying Class-Based Active EEM Policies

To verify the active or the running EEM policies, use the **show event manager policy active** command.

### SUMMARY STEPS

1. **show event manager policy active** [*queue-type* {*applet* | *call-home* | *axp* | *script*}] *class class-options* | *detailed*]

### DETAILED STEPS

---

**show event manager policy active** [*queue-type* {*applet* | *call-home* | *axp* | *script*}] *class class-options* | *detailed*]

This command displays only the running EEM policies. This command includes **class**, **detailed** and **queue-type** optional keywords. The following is sample output from this command:

#### Example:

```
Device# show event manager policy active
no. job id p s status time of event event type name
1 12598 N A running Mon Oct29 20:49:37 2007 timer watchdog loop.tcl
2 12609 N A running Mon Oct29 20:49:42 2007 timer watchdog loop.tcl
3 12620 N A running Mon Oct29 20:49:46 2007 timer watchdog loop.tcl
4 12650 N A running Mon Oct29 20:49:59 2007 timer watchdog loop.tcl
5 12842 N A running Mon Oct29 20:51:13 2007 timer watchdog loop.tcl
default class - 6 applet events
no. job id p s status time of event event type name
1 15852 N A running Mon Oct29 21:11:09 2007 counter WDOG_SYSLG_CNTR_TRACK_INTF_APPL
2 15853 N A running Mon Oct29 21:11:09 2007 counter WDOG_SYSLG_CNTR_TRACK_INTF_APPL
3 15854 N A running Mon Oct29 21:11:10 2007 counter WDOG_SYSLG_CNTR_TRACK_INTF_APPL
4 15855 N A running Mon Oct29 21:11:10 2007 timer watchdog WDOG_SYSLG_CNTR_TRACK_INTF_APPL
```

```

5 15856 N A running Mon Oct29 21:11:11 2007 counter WDOG_SYSLG_CNTR_TRACK_INTF_APPL
6 15858 N A running Mon Oct29 21:11:11 2007 counter WDOG_SYSLG_CNTR_TRACK_INTF_APPL

```

## Verifying Class-Based Active EEM Policies

To verify the active or the running EEM policies, use the **show event manager policy active** command.

### SUMMARY STEPS

1. **show event manager policy active** [*queue-type* {*applet*|*call-home* | *axp* | *script*}] *class class-options* | *detailed*]

### DETAILED STEPS

**show event manager policy active** [*queue-type* {*applet*|*call-home* | *axp* | *script*}] *class class-options* | *detailed*]

This command displays only the running EEM policies. This command includes **class**, **detailed** and **queue-type** optional keywords. The following is sample output from this command:

#### Example:

```

Device# show event manager policy active
no. job id p s status time of event event type name
1 12598 N A running Mon Oct29 20:49:37 2007 timer watchdog loop.tcl
2 12609 N A running Mon Oct29 20:49:42 2007 timer watchdog loop.tcl
3 12620 N A running Mon Oct29 20:49:46 2007 timer watchdog loop.tcl
4 12650 N A running Mon Oct29 20:49:59 2007 timer watchdog loop.tcl
5 12842 N A running Mon Oct29 20:51:13 2007 timer watchdog loop.tcl
default class - 6 applet events
no. job id p s status time of event event type name
1 15852 N A running Mon Oct29 21:11:09 2007 counter WDOG_SYSLG_CNTR_TRACK_INTF_APPL
2 15853 N A running Mon Oct29 21:11:09 2007 counter WDOG_SYSLG_CNTR_TRACK_INTF_APPL
3 15854 N A running Mon Oct29 21:11:10 2007 counter WDOG_SYSLG_CNTR_TRACK_INTF_APPL
4 15855 N A running Mon Oct29 21:11:10 2007 timer watchdog WDOG_SYSLG_CNTR_TRACK_INTF_APPL
5 15856 N A running Mon Oct29 21:11:11 2007 counter WDOG_SYSLG_CNTR_TRACK_INTF_APPL
6 15858 N A running Mon Oct29 21:11:11 2007 counter WDOG_SYSLG_CNTR_TRACK_INTF_APPL

```

## Verifying Pending EEM Policies

To verify the EEM policies that are pending for execution, use the **show event manager policy pending** command. Use the optional keywords to specify EEM class-based scheduling options.

### SUMMARY STEPS

1. **show event manager policy pending** [*queue-type* {*applet*|*call-home* | *axp* | *script*}] *class class-options* | *detailed*]

### DETAILED STEPS

**show event manager policy pending** [*queue-type* {*applet*|*call-home* | *axp* | *script*}] *class class-options* | *detailed*]

This command displays only the pending policies. This command includes **class**, **detailed** and **queue-type** optional keywords. The following is sample output from this command:

**Example:**

```
Device# show event manager policy pending
no. job id p s status time of event event type name
1 12851 N A pend Mon Oct29 20:51:18 2007 timer watchdog loop.tcl
2 12868 N A pend Mon Oct29 20:51:24 2007 timer watchdog loop.tcl
3 12873 N A pend Mon Oct29 20:51:27 2007 timer watchdog loop.tcl
4 12907 N A pend Mon Oct29 20:51:41 2007 timer watchdog loop.tcl
5 13100 N A pend Mon Oct29 20:52:55 2007 timer watchdog loop.tcl
```

## Configuring EEM Applet (Interactive CLI) Support

The synchronous applets are enhanced to support interaction with the local console (tty) using two commands, **action gets** and **action puts**, and these commands allow users to enter and display input directly on the console. The output for synchronous applets will bypass the system logger. The local console will be opened by the applets and serviced by the corresponding synchronous Event Detector pty. Synchronous output will be directed to the opened console.

### Reading and Writing Input from the Active Console for Synchronous EEM Applets

Use the following tasks to implement EEM applet interactive CLI support:

#### Reading Input from the Active Console

When a synchronous policy is triggered, the related console is stored in the publish information specification. The policy director will query this information in an event\_reqinfo call, and store the given console information for use by the **action gets** command.

The **action gets** command reads a line of the input from the active console and stores the input in the variable. The trailing new line will not be returned.

#### SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **event manager applet** *applet-name*
4. **event none**
5. **action** *label* **gets** *variable*
6. **action** *label* **syslog** [**priority** *priority-level*] **msg** *msg-text*
7. **exit**

#### DETAILED STEPS

|        | Command or Action                    | Purpose                                                                 |
|--------|--------------------------------------|-------------------------------------------------------------------------|
| Step 1 | <b>enable</b><br><br><b>Example:</b> | Enables privileged EXEC mode.<br><br>• Enter your password if prompted. |

|               | Command or Action                                                                                                                                                                       | Purpose                                                                                                                                                                                                                           |
|---------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|               | Device> enable                                                                                                                                                                          |                                                                                                                                                                                                                                   |
| <b>Step 2</b> | <b>configure terminal</b><br><b>Example:</b><br>Device# configure terminal                                                                                                              | Enters global configuration mode.                                                                                                                                                                                                 |
| <b>Step 3</b> | <b>event manager applet</b> <i>applet-name</i><br><b>Example:</b><br>Device(config)# event manager applet action                                                                        | Registers the applet with the EEM and enters applet configuration mode.                                                                                                                                                           |
| <b>Step 4</b> | <b>event none</b><br><b>Example:</b><br>Device(config-applet)# event none                                                                                                               | Specifies that an EEM policy is to be registered with the EEM and can be run manually.                                                                                                                                            |
| <b>Step 5</b> | <b>action</b> <i>label gets variable</i><br><b>Example:</b><br>Device(config-applet)# action label2 gets input                                                                          | Gets input from the local console in a synchronous applet and stores the value in the given variable when an EEM applet is triggered.                                                                                             |
| <b>Step 6</b> | <b>action</b> <i>label syslog [priority priority-level] msg msg-text</i><br><b>Example:</b><br>Device(config-applet)# action label3 syslog msg "Input entered was \" <i>\$input</i> \"" | Specifies the action to be taken when an EEM applet is triggered. <ul style="list-style-type: none"> <li>In this example, the action to be taken is to write the value of the variable specified in Step 5, to syslog.</li> </ul> |
| <b>Step 7</b> | <b>exit</b><br><b>Example:</b><br>Device(config-applet)# exit                                                                                                                           | Exits applet configuration mode and returns to privileged EXEC mode.                                                                                                                                                              |

### Example

The following example shows how to get the input from the local tty in a synchronous applet and store the value

```
Device(config)# event manager applet action
Device(config-applet)# event none
Device(config-applet)# action label2 gets input
Device(config-applet)# action label3 syslog msg "Input entered was \"$input\""
```

## Writing Input to the Active Console

When a synchronous policy is triggered, the related console is stored in the publish information specification. The policy director will query this information in an event\_reqinfo call, and store the given console information for use by the **action puts** command.

The **action puts** command will write the string to the active console. A new line will be displayed unless the **newline** keyword is specified. The output from the **action puts** command for a synchronous applet is displayed directly to the console, bypassing the system logger. The output of the **action puts** command for an asynchronous applet is directed to the system logger.

### SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **event manager applet** *applet-name*
4. **event none**
5. **action** *label* **regexp** *string-pattern string-input* [*string-match* [*string-submatch1*] [*string-submatch2*] [*string-submatch3*]]
6. **action** *label* **puts** [**newline**] *string*
7. **exit**
8. **event manager run** *applet-name*

### DETAILED STEPS

|        | Command or Action                                                                                                                                                                            | Purpose                                                                                                            |
|--------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------|
| Step 1 | <b>enable</b><br><b>Example:</b><br>Device> enable                                                                                                                                           | Enables privileged EXEC mode. <ul style="list-style-type: none"> <li>• Enter your password if prompted.</li> </ul> |
| Step 2 | <b>configure terminal</b><br><b>Example:</b><br>Device# configure terminal                                                                                                                   | Enters global configuration mode.                                                                                  |
| Step 3 | <b>event manager applet</b> <i>applet-name</i><br><b>Example:</b><br>Device(config)# event manager applet action                                                                             | Registers the applet with the EEM and enters applet configuration mode.                                            |
| Step 4 | <b>event none</b><br><b>Example:</b><br>Device(config-applet)# event none                                                                                                                    | Specifies that an EEM policy is to be registered with the EEM and can be run manually.                             |
| Step 5 | <b>action</b> <i>label</i> <b>regexp</b> <i>string-pattern string-input</i><br>[ <i>string-match</i> [ <i>string-submatch1</i> ] [ <i>string-submatch2</i> ]<br>[ <i>string-submatch3</i> ]] | Specifies the action to match the regular expression pattern on an input string when an EEM applet is triggered.   |

|               | Command or Action                                                                                                                                                       | Purpose                                                                                                                                                                                                                                                             |
|---------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|               | <p><b>Example:</b></p> <pre>Device(config-applet)# action 1 regexp "(.*) (.*) (.*)" "one two three" _match _sub1</pre>                                                  |                                                                                                                                                                                                                                                                     |
| <b>Step 6</b> | <p><b>action</b> <i>label</i> <b>puts</b> [<b>newline</b>] <i>string</i></p> <p><b>Example:</b></p> <pre>Device(config-applet)# action 2 puts "match is \$_match"</pre> | <p>Specifies the action of printing data directly to the local console when an EEM applet is triggered.</p> <ul style="list-style-type: none"> <li>The <b>newline</b> keyword is optional and is used to suppress the display of the new line character.</li> </ul> |
| <b>Step 7</b> | <p><b>exit</b></p> <p><b>Example:</b></p> <pre>Device(config-applet)# exit</pre>                                                                                        | <p>Exits applet configuration mode and returns to privileged EXEC mode.</p>                                                                                                                                                                                         |
| <b>Step 8</b> | <p><b>event manager run</b> <i>applet-name</i></p> <p><b>Example:</b></p> <pre>Device# event manager run action</pre>                                                   | <p>Manually runs a registered EEM policy.</p> <ul style="list-style-type: none"> <li>In this example, the policy registered in Step 3 is triggered and the associated actions specified in Step 5 and Step 6 are executed.</li> </ul>                               |

### Example

The following example shows how the **action puts** command prints data directly to the local console:

```
Device(config-applet)# event manager applet puts
Device(config-applet)# event none
Device(config-applet)# action 1 regexp "(.*) (.*) (.*)" "one two three" _match _sub1
Device(config-applet)# action 2 puts "match is $_match"
Device(config-applet)# action 3 puts "submatch 1 is $_sub1"
Device# event manager run puts
match is one two three
submatch 1 is one
```

## Configuring SNMP Library Extensions

Depending on your release, the SNMP Library Extensions feature allows you to perform the following configurations.

### Prerequisites

To use this feature, you must be running Cisco IOS Release 12.4(22)T or a later release.

### SNMP Get and Set Operations

The SNMP Library Extensions feature extends the EEM applet **action info** and Tcl **sys\_reqinfo\_snmp** commands to include functionality for SNMP get-one, get-next, getid and set-any operations.



## SNMP Get Operation

The SNMP event manager performs the SNMP get operation to retrieve one or more variables for the managed objects. Using the **action info type snmp oid get-type** and **action info type snmp getid** commands, you can configure the SNMP event manager to send an SNMP get request by specifying the variables to retrieve, and the IP address of the agent.

For example, if you want to retrieve the variable with the OID value of 1.3.6.1.2.1.1.1, you should specify the variable value, that is 1.3.6.1.2.1.1.1. If the specified values do not match, a trap will be generated and an error message will be written to the syslog history.

The **action info type snmp oid get-type** command specifies the type of the get operation to be performed. To retrieve the exact variable, the get operation type should be specified as **exact**. To retrieve a lexicographical successor of the specified OID value, the get operation type should be set to **next**.

The table below shows the built-in variables, in which the values retrieved from SNMP get operation are stored.

*Table 199: Built-in Variables for action info type snmp oid Command*

| Built-in Variable       | Description                                           |
|-------------------------|-------------------------------------------------------|
| <b>_info_snmp_oid</b>   | The SNMP object ID.                                   |
| <b>_info_snmp_value</b> | The value string of the associated SNMP data element. |

## GetID Operation

The **action info type snmp getid** command retrieves the following variables from the SNMP entity:

- sysDescr.0
- sysObjectID.0
- sysUpTime.0
- sysContact.0
- sysName.0
- sysLocation.0

The table below shows the built-in variables, in which the values retrieved from the SNMP getID operation are stored.

*Table 200: Built-in Variables for action info type snmp getid Command*

| Built-in Variable                   | Description                                    |
|-------------------------------------|------------------------------------------------|
| <b>_info_snmp_syslocation_oid</b>   | The OID value of the sysLocation variable.     |
| <b>_info_snmp_syslocation_value</b> | The value string for the sysLocation variable. |
| <b>_info_snmp_sysdescr_oid</b>      | The OID value of the sysDescr variable.        |
| <b>_info_snmp_sysdescr_value</b>    | The value string for the sysDescr variable.    |
| <b>_info_snmp_sysobjectid_oid</b>   | The OID value of the sysObjectID variable.     |

| Built-in Variable                         | Description                                    |
|-------------------------------------------|------------------------------------------------|
| <code>_info_snmp_sysobjectid_value</code> | The value string for the sysObjectID variable. |
| <code>_info_snmp_sysuptime_oid</code>     | The OID value of the sysUptime variable.       |
| <code>_info_snmp_sysuptime_value</code>   | The value string for the sysUptime variable.   |
| <code>_info_snmp_syscontact_oid</code>    | The OID value of the sysContact variable.      |
| <code>_info_snmp_syscontact_value</code>  | The value string for the sysContact variable.  |

The get operation requests can be sent to both local and remote hosts.

## SNMP Set Operation

All SNMP variables are assigned a default value in the MIB view. The SNMP event manager can modify the value of these MIB variables through set operation. The set operation can be performed only on the system that allows read-write access.

To perform a set operation, you must specify the type of the variable and the value associated with it.

The table below shows the valid OID types and values for each OID type.

**Table 201: OID Type and Value for Set Operation**

| OID Type            | Description                                                                                                                                                                                                                                                |
|---------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>counter32</b>    | A 32-bit number with a minimum value of 0. When the maximum value is reached, the counter resets to 0. Integer value in the range from 0 to 4294967295 is valid.                                                                                           |
| <b>gauge</b>        | A 32-bit number with a minimum value of 0. For example, the interface speed on a device is measured using a gauge object type. Integer value in the range from 0 to 4294967295 is valid.                                                                   |
| <b>integer</b>      | A 32-bit number used to specify a numbered type within the context of a managed object. For example, to set the operational status of a device interface, 1 represents up and 2 represents down. Integer value in the range from 0 to 4294967295 is valid. |
| <b>ipv4</b>         | IP version 4 address. IPv4 address in dotted decimal notation is valid.                                                                                                                                                                                    |
| <b>octet string</b> | An octet string in hexadecimal notation used to represent physical addresses. Text strings are valid.                                                                                                                                                      |
| <b>string</b>       | An octet string in text notation used to represent text strings. Text strings are valid.                                                                                                                                                                   |

| OID Type   | Description                                                                                                         |
|------------|---------------------------------------------------------------------------------------------------------------------|
| unsigned32 | A 32-bit number used to represent decimal value. Unsigned integer value in the range from 0 to 4294967295 is valid. |

The set operation can be carried out on both local and remote hosts.

## SNMP Traps and Inform Requests

Traps are SNMP notifications that alert the SNMP manager or the NMS to a network condition.

SNMP inform requests refer to the SNMP notifications that alert the SNMP manager to a network condition and request for confirmation of receipt from the SNMP manager.

An SNMP event occurs when SNMP MIB object ID values are sampled, or when the SNMP counter crosses a defined threshold. If the notifications are enabled and configured for such events, the SNMP traps or inform messages are generated. An SNMP notification event is triggered when an SNMP trap or inform message is received by the event manager server.

To send an SNMP trap or inform message when an Embedded Event Manager (EEM) applet is triggered, the **action info type snmp trap** and **action info type snmp inform** commands are used. The CISCO-EMBEDDED-EVENT-MGR-MIB.mib is used to define the trap and inform messages.

## Configuring EEM Applet for SNMP Get and Set Operations

While registering a policy with the event manager server, the actions associated with an SNMP event can be configured.

Perform this task to configure EEM applet for SNMP set and get operations.

### Before you begin

- SNMP event manager must be configured using the **snmp-server manager** command.
- The SNMP community string should be set by using the **snmp-server community** command to enable access to the SNMP entity.

### SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **event manager applet** *applet-name*
4. Do one of the following:
  - **event snmp oid** *oid-value* **get-type** {**exact** | **next**} **entry-op** *operator* **entry-val** *entry-value* [**exit-comb** | **and**] [**exit-op** *operator*] [**exit-val** *exit-value*] [**exit-time** *exit-time-value*] **poll-interval** *poll-int-value*
5. **action label info type snmp oid** *oid-value* **get-type** {**exact** | **next**} [**community** *community-string*] [**ipaddr** *ip-address*]
6. **action label info type snmp oid** *oid-value* **set-type** *oid-type* *oid-type-value* **community** *community-string* [**ipaddr** *ip-address*]
7. **action label info type snmp getid** *oid-value* [**community** *community-string*] [**ipaddr** *ip-address*]

## 8. exit

## DETAILED STEPS

|               | Command or Action                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 | Purpose                                                                                                                                                                                                                                                                                                                  |
|---------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Step 1</b> | <b>enable</b><br><b>Example:</b><br>Device> enable                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                | Enables privileged EXEC mode. <ul style="list-style-type: none"> <li>• Enter your password if prompted.</li> </ul>                                                                                                                                                                                                       |
| <b>Step 2</b> | <b>configure terminal</b><br><b>Example:</b><br>Device# configure terminal                                                                                                                                                                                                                                                                                                                                                                                                                                                                        | Enters global configuration mode.                                                                                                                                                                                                                                                                                        |
| <b>Step 3</b> | <b>event manager applet <i>applet-name</i></b><br><b>Example:</b><br>Device(config)# event manager applet snmp                                                                                                                                                                                                                                                                                                                                                                                                                                    | Registers the applet with the event manager server and enters applet configuration mode.                                                                                                                                                                                                                                 |
| <b>Step 4</b> | Do one of the following: <ul style="list-style-type: none"> <li>• <b>event snmp oid <i>oid-value</i> get-type {exact   next} entry-op <i>operator</i> entry-val <i>entry-value</i>[exit-comb   and}] [exit-op <i>operator</i>] [exit-val <i>exit-value</i>] [exit-time <i>exit-time-value</i>] poll-interval <i>poll-int-value</i></b></li> </ul> <b>Example:</b><br>Device(config-applet)# event snmp oid<br><b>Example:</b><br>1.3.6.1.4.1.9.9.48.1.1.1.6.1 get-type exact<br><b>Example:</b><br>entry-op lt entry-val 5120000 poll-interval 90 | Specifies the event criteria that cause the EEM applet to run. <ul style="list-style-type: none"> <li>• In this example, an EEM event is triggered when free memory falls below the value of 5120000.</li> <li>• Exit criteria are optional, and if not specified, event monitoring is reenabled immediately.</li> </ul> |
| <b>Step 5</b> | <b>action label info type snmp oid <i>oid-value</i> get-type {exact  next} [community <i>community-string</i>] [ipaddr <i>ip-address</i>]</b><br><b>Example:</b><br>Device(config-applet)# action 1.3 info type<br><b>Example:</b><br>snmp oid 1.3.6.1.4.1.9.9.48.1.1.1.6.1 get-type<br><b>Example:</b><br>exact community public ipaddr 172.17.16.69                                                                                                                                                                                             | Specifies the type of get operation to perform. <ul style="list-style-type: none"> <li>• In this example, the type of get operation is specified as exact and community string is specified as public.</li> </ul>                                                                                                        |

|        | Command or Action                                                                                                                                                                                                                                                                                                                                                                                                                                                      | Purpose                                                                                                                                                                                                                                                                                                     |
|--------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Step 6 | <p><b>action label info type snmp oid <i>oid-value</i> set-type <i>oid-type</i> <i>oid-type-value</i> community <i>community-string</i> [<i>ipaddr ip-address</i>]</b></p> <p><b>Example:</b></p> <pre>Device(config-applet)# action 1.4 info type</pre> <p><b>Example:</b></p> <pre>snmp oid 1.3.6.1.4.1.9.9.48.1.1.1.6.1 set-type</pre> <p><b>Example:</b></p> <pre>integer 42220 sysName.0 community rw ipaddr</pre> <p><b>Example:</b></p> <pre>172.17.16.69</pre> | <p>(Optional) Specifies the variable to be set.</p> <ul style="list-style-type: none"> <li>In this example, the sysName.0 variable is specified for the set operation and community string is specified as rw.</li> </ul> <p><b>Note</b> For set operation, you must specify the SNMP community string.</p> |
| Step 7 | <p><b>action label info type snmp getid <i>oid-value</i> [<i>community</i> <i>community-string</i>] [<i>ipaddr ip-address</i>]</b></p> <p><b>Example:</b></p> <pre>Device(config-applet)# action 1.3 info type</pre> <p><b>Example:</b></p> <pre>snmp getid community public ipaddr 172.17.16.69</pre>                                                                                                                                                                 | <p>(Optional) Specifies if the individual variables should be retrieved by the getid operation.</p>                                                                                                                                                                                                         |
| Step 8 | <p><b>exit</b></p> <p><b>Example:</b></p> <pre>Device(config)# exit</pre>                                                                                                                                                                                                                                                                                                                                                                                              | <p>Exits global configuration mode and returns to privileged EXEC mode.</p>                                                                                                                                                                                                                                 |

## Configuring EEM Applet for SNMP OID Notifications

Perform this task to configure SNMP notifications.

### Before you begin

- SNMP event manager must be configured using the **snmp-server manager** command and SNMP agents must be configured to send and receive SNMP traps generated for an EEM policy.
- SNMP traps and informs must be enabled by using the **snmp-server enable traps event-manager** and **snmp-server enable traps** commands, to allow traps and inform requests to be sent from the device to the event manager server.

### SUMMARY STEPS

- enable
- configure terminal

3. **event manager applet** *applet-name*
4. Do one of the following:
  - **event snmp oid** *oid-value* **get-type** {**exact** | **next**} **entry-op** *operator* **entry-val** *entry-value* [**exit-comb** | **and**] [**exit-op** *operator*] [**exit-val** *exit-value*] [**exit-time** *exit-time-value*] **poll-interval** *poll-int-value*
5. **action label info type snmp var** *variable-name* **oid** *oid-value* *oid-type* *oid-type-value*
6. **action label info type snmp trap enterprise-oid** *enterprise-oid-value* **generic-trapnum** *generic-trap-number* **specific-trapnum** *specific-trap-number* **trap-oid** *trap-oid-value* **trap-var** *trap-variable*
7. **action label info type snmp inform trap-oid** *trap-oid-value* **trap-var** *trap-variable* **community** *community-string* **ipaddr** *ip-address*
8. **exit**

## DETAILED STEPS

|               | Command or Action                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              | Purpose                                                                                                                                                                                                                                                                                                                  |
|---------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Step 1</b> | <b>enable</b><br><b>Example:</b><br>Device> enable                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             | Enables privileged EXEC mode. <ul style="list-style-type: none"> <li>• Enter your password if prompted.</li> </ul>                                                                                                                                                                                                       |
| <b>Step 2</b> | <b>configure terminal</b><br><b>Example:</b><br>Device# configure terminal                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     | Enters global configuration mode.                                                                                                                                                                                                                                                                                        |
| <b>Step 3</b> | <b>event manager applet</b> <i>applet-name</i><br><b>Example:</b><br>Device(config)# event manager applet snmp                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 | Registers the applet with the event manager server and enters applet configuration mode.                                                                                                                                                                                                                                 |
| <b>Step 4</b> | Do one of the following: <ul style="list-style-type: none"> <li>• <b>event snmp oid</b> <i>oid-value</i> <b>get-type</b> {<b>exact</b>   <b>next</b>} <b>entry-op</b> <i>operator</i> <b>entry-val</b> <i>entry-value</i> [<b>exit-comb</b>   <b>and</b>] [<b>exit-op</b> <i>operator</i>] [<b>exit-val</b> <i>exit-value</i>] [<b>exit-time</b> <i>exit-time-value</i>] <b>poll-interval</b> <i>poll-int-value</i></li> </ul> <b>Example:</b><br>Device(config-applet)# event snmp oid<br><b>Example:</b><br>1.3.6.1.4.1.9.9.48.1.1.1.6.1 get-type exact<br><b>Example:</b><br>entry-op lt entry-val 5120000 poll-interval 90 | Specifies the event criteria that cause the EEM applet to run. <ul style="list-style-type: none"> <li>• In this example, an EEM event is triggered when free memory falls below the value of 5120000.</li> <li>• Exit criteria are optional, and if not specified, event monitoring is reenabled immediately.</li> </ul> |

|        | Command or Action                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        | Purpose                                                                                                                                                                                                                                                                                                                                                                                                                                  |
|--------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Step 5 | <p><b>action label info type snmp var</b> <i>variable-name oid oid-value oid-type oid-type-value</i></p> <p><b>Example:</b></p> <pre>Device(config-applet)# action 1.3 info type</pre> <p><b>Example:</b></p> <pre>snmp var sysDescr.0 oid</pre> <p><b>Example:</b></p> <pre>1.3.6.1.4.1.9.9.48.1.1.1.6.1 integer 4220</pre>                                                                                                                                                                                                             | <p>Specifies the instance of a managed object and its value.</p> <ul style="list-style-type: none"> <li>In this example, the sysDescr.0 variable is used.</li> </ul>                                                                                                                                                                                                                                                                     |
| Step 6 | <p><b>action label info type snmp trap enterprise-oid</b> <i>enterprise-oid-value generic-trapnum generic-trap-number specific-trapnum specific-trap-number trap-oid trap-oid-value trap-var trap-variable</i></p> <p><b>Example:</b></p> <pre>Device(config-applet)# action 1.4 info type</pre> <p><b>Example:</b></p> <pre>snmp trap enterprise-oid 1.3.6.1.4.1.1</pre> <p><b>Example:</b></p> <pre>generic-trapnum 4 specific-trapnum 7 trap-oid</pre> <p><b>Example:</b></p> <pre>1.3.6.1.4.1.1.226.0.2.1 trap-var sysUpTime.0</pre> | <p>Generates an SNMP trap when the EEM applet is triggered.</p> <ul style="list-style-type: none"> <li>In this example, the authenticationFailure trap is generated.</li> </ul> <p><b>Note</b> The specific trap number refers to the enterprise-specific trap, which is generated when an enterprise event occurs. If the generic trap number is not set to 6, the specific trap number you specify will be used to generate traps.</p> |
| Step 7 | <p><b>action label info type snmp inform trap-oid</b> <i>trap-oid-value trap-var trap-variable community community-string ipaddr ip-address</i></p> <p><b>Example:</b></p> <pre>Device(config-applet)# action 1.4 info type</pre> <p><b>Example:</b></p> <pre>snmp inform trap-oid 1.3.6.1.4.1.1.226.0.2.1</pre> <p><b>Example:</b></p> <pre>trap-var sysUpTime.0 community public ipaddr</pre> <p><b>Example:</b></p> <pre>172.69.16.2</pre>                                                                                            | <p>Generates an SNMP inform request when the EEM applet is triggered.</p> <ul style="list-style-type: none"> <li>In this example, the inform request is generated for the sysUpTime.0 variable.</li> </ul>                                                                                                                                                                                                                               |

|               | Command or Action                                          | Purpose                                                         |
|---------------|------------------------------------------------------------|-----------------------------------------------------------------|
| <b>Step 8</b> | <b>exit</b><br><b>Example:</b><br><br>Device(config)# exit | Exits global configuration mode and returns to privileged mode. |

## Configuring Variable Logic for EEM Applets

The Variable Logic for EEM Applets feature adds the ability to apply conditional logic within EEM applets. Before variable logic is introduced, applets have a linear structure where each action is executed in the order in which they are configured when the event is triggered. Conditional logic introduces a control structure that can change the flow of actions within applets depending on conditional expressions. Each control structure can contain a list of applet actions including looping and if/else actions which determine if the structure is executed or not.

The information in applet configuration mode is presented as background to set the context for the action commands.

To provide a consistent user interface between the Tool Command Language (Tcl) and the applet (CLI) based EEM policies, the following criteria are followed:

- Event specification criteria are written in Tcl in the Tcl based implementation.
- Event specification data is written using the CLI applet submode configuration statements in the applet-based implementation.

Applet configuration mode is entered using the event manager applet command. In applet configuration mode the config prompt changes to (config-applet)#. In applet configuration mode two types of config statements are supported:

- event - used to specify the event criteria to cause this applet to run.
- action - used to specify a built-in action to perform.

Multiple **action** applet config commands are allowed within an applet configuration. If no **action** applet config command is present, a warning is displayed, upon exit, stating no statements are associated with this applet. When no statements are associated with this applet, events get triggered but no action is taken. If no commands are specified in applet configuration mode, the applet will be removed upon exit. The exit applet config command is used to exit from applet configuration mode.

Depending on your release, the Variable Logic for EEM Applets feature allows you to perform the following configurations.

### Prerequisites

To use this feature, you must be running Cisco IOS Release 12.4(22)T or a later release.

### Configuring Variable Logic for EEM Applets

EEM 3.0 adds new applet action commands to permit simple variable logic within applets.

To configure the variable logic using action commands perform the following tasks.



## Specifying a Loop of Conditional Blocks

To specify a loop of a conditional block when an EEM applet is triggered, perform this task. In this task, a conditional loop is set to check if the value of the variable is less than 10. If the value of the variable is less than 10, then the message 'i is \$\_i' is written to the syslog.



**Note** Depending on your release, the **set** (EEM) command is replaced by the **action set** command. See the **action label set** command for more information. If the set (EEM) command is entered in certain releases, the IOS parser translates the **set** command to the **action label set** command.

### SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **event manager applet** *applet-name*
4. **action label set**
5. **action label while** *string\_op1 operator string\_op2*
6. Add any action as required.
7. **action label end**

### DETAILED STEPS

|        | Command or Action                                                                                                   | Purpose                                                                                                                                                                              |
|--------|---------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Step 1 | <b>enable</b><br><b>Example:</b><br>Device> enable                                                                  | Enables privileged EXEC mode. <ul style="list-style-type: none"> <li>• Enter your password if prompted.</li> </ul>                                                                   |
| Step 2 | <b>configure terminal</b><br><b>Example:</b><br>Device# configure terminal                                          | Enters global configuration mode.                                                                                                                                                    |
| Step 3 | <b>event manager applet</b> <i>applet-name</i><br><b>Example:</b><br>Device(config)# event manager applet condition | Registers the applet with the Embedded Event Manager (EEM) and enters applet configuration mode.                                                                                     |
| Step 4 | <b>action label set</b><br><b>Example:</b><br>Device(config-applet)# <b>action 1.0 set i 2</b>                      | Sets an action for the event. <ul style="list-style-type: none"> <li>• In this example, the value of the variable i is set to 2.</li> </ul>                                          |
| Step 5 | <b>action label while</b> <i>string_op1 operator string_op2</i><br><b>Example:</b>                                  | Specifies a loop of a conditional block. <ul style="list-style-type: none"> <li>• In this example, a loop is set to check if the value of the variable i is less than 10.</li> </ul> |

|               | Command or Action                                                                                                          | Purpose                                                                                                                                                                             |
|---------------|----------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|               | <code>Device(config-applet)# action 2 while \$i lt 10</code>                                                               |                                                                                                                                                                                     |
| <b>Step 6</b> | <p>Add any action as required.</p> <p><b>Example:</b></p> <pre>Device(config-applet)# action 3 syslog msg "i is \$i"</pre> | <p>Performs the action as indicated by the action command.</p> <ul style="list-style-type: none"> <li>In this example, the message 'i is \$_i' is written to the syslog.</li> </ul> |
| <b>Step 7</b> | <p><code>action label end</code></p> <p><b>Example:</b></p> <pre>Device(config-applet)# action 3 end</pre>                 | Exits from the running action.                                                                                                                                                      |

## Specifying if else Conditional Blocks

To specify the beginning of an if conditional statement followed by an else conditional statement, perform this task. The if or else conditional statements can be used in conjunction with each other or separately. In this task, the value of a variable is set to 5. An if conditional block is then specified to check if the value of the variable is less than 10. Provided the if conditional block is satisfied, an action command to output the message 'x is less than 10' is specified.

Following the if conditional block, an else conditional block is specified. Provided the if conditional block is not satisfied, an action command to output the message 'x is greater than 10' is specified.

### SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **event manager applet** *applet-name*
4. **action label set** *variable-name variable-value*
5. **action label if** [*stringop1*] {**eq** | **gt** | **ge** | **lt** | **le** | **ne**} [*stringop2*]
6. Add any action as required.
7. **action label else**
8. Add any action as required.
9. **end**

### DETAILED STEPS

|               | Command or Action                                                        | Purpose                                                                                                                 |
|---------------|--------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------|
| <b>Step 1</b> | <p><b>enable</b></p> <p><b>Example:</b></p> <pre>Device&gt; enable</pre> | <p>Enables privileged EXEC mode.</p> <ul style="list-style-type: none"> <li>Enter your password if prompted.</li> </ul> |
| <b>Step 2</b> | <p><b>configure terminal</b></p> <p><b>Example:</b></p>                  | Enters global configuration mode.                                                                                       |

|               | Command or Action                                                                                                                                                                                                                          | Purpose                                                                                                                                                                                       |
|---------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|               | <code>Device# configure terminal</code>                                                                                                                                                                                                    |                                                                                                                                                                                               |
| <b>Step 3</b> | <b>event manager applet</b> <i>applet-name</i><br><b>Example:</b><br><code>Device(config)# event manager applet ifcondition</code>                                                                                                         | Registers the applet with the Embedded Event Manager (EEM) and enters applet configuration mode.                                                                                              |
| <b>Step 4</b> | <b>action</b> <i>label</i> <b>set</b> <i>variable-name</i> <i>variable-value</i><br><b>Example:</b><br><code>Device(config-applet)# action 1.0 set x 5</code>                                                                              | Sets an action for the event.<br><ul style="list-style-type: none"><li>In this example, the value of the variable x is set to 5.</li></ul>                                                    |
| <b>Step 5</b> | <b>action</b> <i>label</i> <b>if</b> [ <i>stringop1</i> ] { <b>eq</b>   <b>gt</b>   <b>ge</b>   <b>lt</b>   <b>le</b>   <b>ne</b> } [ <i>stringop2</i> ]<br><b>Example:</b><br><code>Device(config-applet)# action 2.0 if \$x lt 10</code> | Specifies an if conditional statement.<br><ul style="list-style-type: none"><li>In this example, an if conditional statement to check if the value of the variable is less than 10.</li></ul> |
| <b>Step 6</b> | Add any action as required.<br><b>Example:</b><br><code>Device(config-applet)# action 3.0 puts "\$x is less than 10"</code>                                                                                                                | Performs the action as indicated by the action command.<br><ul style="list-style-type: none"><li>In this example, the message '5 is less than 10' is displayed on the screen.</li></ul>       |
| <b>Step 7</b> | <b>action</b> <i>label</i> <b>else</b><br><b>Example:</b><br><code>Device(config-applet)# action 4.0 else</code>                                                                                                                           | Specifies an else conditional statement                                                                                                                                                       |
| <b>Step 8</b> | Add any action as required.<br><b>Example:</b><br><code>Device(config-applet)# action 5.0</code>                                                                                                                                           | Performs the action as indicated by the action command.<br><ul style="list-style-type: none"><li>In this example, the message '5 is greater than 10' is displayed on the screen.</li></ul>    |
| <b>Step 9</b> | <b>end</b><br><b>Example:</b><br><code>Device(config-applet)# end</code>                                                                                                                                                                   | Exits from the running action.                                                                                                                                                                |

## Specifying foreach Iterating Statements

To specify a conditional statement that iterates over an input string using the delimiter as a tokenizing pattern, perform this task. The foreach iteration statement is used to iterate through a collection to get the desired information. The delimiter is a regular expression pattern string. The token found in each iteration is assigned to the given iterator variable. All arithmetic calculations are performed as long integers with out any checks for overflow. In this task, the value of the variable x is set to 5. An iteration statement is set to run through

the input string red, blue, green, orange. For every element in the input string, a corresponding message is displayed on the screen.

## SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **event manager applet** *applet-name*
4. **action label foreach** [*string-iterator*] [*string-input*] [*string-delimiter*]
5. Specify any action command
6. **action label end**

## DETAILED STEPS

|               | Command or Action                                                                                                                                                                                           | Purpose                                                                                                                                                                                                                                                       |
|---------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Step 1</b> | <b>enable</b><br><b>Example:</b><br>Device> enable                                                                                                                                                          | Enables privileged EXEC mode. <ul style="list-style-type: none"> <li>• Enter your password if prompted.</li> </ul>                                                                                                                                            |
| <b>Step 2</b> | <b>configure terminal</b><br><b>Example:</b><br>Device# configure terminal                                                                                                                                  | Enters global configuration mode.                                                                                                                                                                                                                             |
| <b>Step 3</b> | <b>event manager applet</b> <i>applet-name</i><br><b>Example:</b><br>Device(config)# event manager applet iteration                                                                                         | Registers the applet with the Embedded Event Manager (EEM) and enters applet configuration mode.                                                                                                                                                              |
| <b>Step 4</b> | <b>action label foreach</b> [ <i>string-iterator</i> ] [ <i>string-input</i> ] [ <i>string-delimiter</i> ]<br><b>Example:</b><br>Device(config-applet)# action 2.0 foreach iterator "red blue green orange" | Iterates over an input string using the delimiter as a tokenizing pattern. <ul style="list-style-type: none"> <li>• In this example, the iteration is run through the elements of the input string - red, blue, green and orange.</li> </ul>                  |
| <b>Step 5</b> | Specify any action command<br><b>Example:</b><br>Device(config-applet)# action 3.0 puts "Iterator is \$iterator"                                                                                            | Performs the action as indicated by the action command. <ul style="list-style-type: none"> <li>• In this example, the following message is displayed on the screen:</li> </ul> Iterator is red<br>Iterator is blue<br>Iterator is green<br>Iterator is orange |

|        | Command or Action                                                                                      | Purpose                        |
|--------|--------------------------------------------------------------------------------------------------------|--------------------------------|
| Step 6 | <p><b>action label end</b></p> <p><b>Example:</b></p> <pre>Device(config-applet)# action 4.0 end</pre> | Exits from the running action. |

## Using Regular Expressions

To match a regular expression pattern with an input string, perform this task. Using regular expressions, you can specify the rules for a set of possible strings to be matched.

### SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **event manager applet** *applet-name*
4. **action label regexp** *string-pattern string-input* [*string-match* [*string-submatch1*] [*string-submatch2*] [*string-submatch3*]]

### DETAILED STEPS

|        | Command or Action                                                                                                                                                                                                                                                                                      | Purpose                                                                                                                                                                                                                                                                                                                                                                                                |
|--------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Step 1 | <p><b>enable</b></p> <p><b>Example:</b></p> <pre>Device&gt; enable</pre>                                                                                                                                                                                                                               | <p>Enables privileged EXEC mode.</p> <ul style="list-style-type: none"> <li>• Enter your password if prompted.</li> </ul>                                                                                                                                                                                                                                                                              |
| Step 2 | <p><b>configure terminal</b></p> <p><b>Example:</b></p> <pre>Device# configure terminal</pre>                                                                                                                                                                                                          | Enters global configuration mode.                                                                                                                                                                                                                                                                                                                                                                      |
| Step 3 | <p><b>event manager applet</b> <i>applet-name</i></p> <p><b>Example:</b></p> <pre>Device(config)# event manager applet<br/>regexp</pre>                                                                                                                                                                | Registers the applet with the Embedded Event Manager (EEM) and enters applet configuration mode.                                                                                                                                                                                                                                                                                                       |
| Step 4 | <p><b>action label regexp</b> <i>string-pattern string-input</i> [<i>string-match</i> [<i>string-submatch1</i>] [<i>string-submatch2</i>] [<i>string-submatch3</i>]]</p> <p><b>Example:</b></p> <pre>Device(config-applet)# action 2.0 regexp "(.*)<br/>(.*) (.*)" "red blue green" _match _sub1</pre> | <p>Specifies an expression pattern to match with an input string.</p> <ul style="list-style-type: none"> <li>• In this example, an input string of ‘red blue green’ is specified. When the expression pattern matches the input string, the entire result <b>red blue green</b> is stored in the variable <b>_match</b> and the submatch <b>red</b> is stored in the variable <b>_sub1</b>.</li> </ul> |

## Incrementing the Values of Variables

To increment the value of variables, perform this task. In this task, the value of a variable is set to 20 and then the value is incremented by 12.

### SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **event manager applet** *applet-name*
4. **action label set**
5. **action label increment** *variable-name long-integer*

### DETAILED STEPS

|               | Command or Action                                                                                                                                   | Purpose                                                                                                                                                                              |
|---------------|-----------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Step 1</b> | <b>enable</b><br><b>Example:</b><br>Device> enable                                                                                                  | Enables privileged EXEC mode. <ul style="list-style-type: none"> <li>• Enter your password if prompted.</li> </ul>                                                                   |
| <b>Step 2</b> | <b>configure terminal</b><br><b>Example:</b><br>Device# configure terminal                                                                          | Enters global configuration mode.                                                                                                                                                    |
| <b>Step 3</b> | <b>event manager applet</b> <i>applet-name</i><br><b>Example:</b><br>Device(config)# event manager applet increment                                 | Registers the applet with the Embedded Event Manager (EEM) and enters applet configuration mode.                                                                                     |
| <b>Step 4</b> | <b>action label set</b><br><b>Example:</b><br>Device(config-applet)# <b>action 1.0 set varname 20</b>                                               | Sets an action for the event. <ul style="list-style-type: none"> <li>• In this example, the value of the variable is set to 20.</li> </ul>                                           |
| <b>Step 5</b> | <b>action label increment</b> <i>variable-name long-integer</i><br><b>Example:</b><br>Device(config-applet)# <b>action 2.0 increment varname 12</b> | Increments the value of variable by the specified long integer. <ul style="list-style-type: none"> <li>• In this example, the value of the variable is incremented by 12.</li> </ul> |

## Configuring Event SNMP Object

Perform this task to register the Simple Network Management Protocol (SNMP) object event for an Embedded Event Manager (EEM) applet that is run by sampling SNMP object.

**SUMMARY STEPS**

1. **enable**
2. **configure terminal**
3. **event manager applet** *applet-name*
4. **event snmp-object oid** *oid-value* **type** *value* **sync** {**yes** | **no**} **skip** {**yes** | **no**} **istable** {**yes** | **no**} [**default** *seconds*] [**maxrun** *maxruntime-number*]
5. **exit**

**DETAILED STEPS**

|               | <b>Command or Action</b>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    | <b>Purpose</b>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |
|---------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Step 1</b> | <p><b>enable</b></p> <p><b>Example:</b></p> <pre>Device&gt; enable</pre>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    | <p>Enables privileged EXEC mode.</p> <ul style="list-style-type: none"> <li>• Enter your password if prompted.</li> </ul>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |
| <b>Step 2</b> | <p><b>configure terminal</b></p> <p><b>Example:</b></p> <pre>Device# configure terminal</pre>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               | <p>Enters global configuration mode.</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |
| <b>Step 3</b> | <p><b>event manager applet</b> <i>applet-name</i></p> <p><b>Example:</b></p> <pre>Device(config)# event manager applet manual-policy</pre>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  | <p>Registers the applet with the Embedded Event Manager and enters applet configuration mode.</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
| <b>Step 4</b> | <p><b>event snmp-object oid</b> <i>oid-value</i> <b>type</b> <i>value</i> <b>sync</b> {<b>yes</b>   <b>no</b>} <b>skip</b> {<b>yes</b>   <b>no</b>} <b>istable</b> {<b>yes</b>   <b>no</b>} [<b>default</b> <i>seconds</i>] [<b>maxrun</b> <i>maxruntime-number</i>]</p> <p><b>Example:</b></p> <pre>Device(config-applet)# event snmp-object oid 1.9.9.9 type gauge sync yes</pre> <p><b>Example:</b></p> <pre>action 1 syslog msg "oid = \$_snmp_oid"</pre> <p><b>Example:</b></p> <pre>action 2 syslog msg "request = \$_snmp_request"</pre> <p><b>Example:</b></p> <pre>action 3 syslog msg "request_type = \$_snmp_request_type"</pre> | <p>Registers the Simple Network Management Protocol (SNMP) object event for an Embedded Event Manager (EEM) applet to intercept SNMP GET and SET requests for an object.</p> <p>The default for this command is that it is not configured. If this command is configured the defaults are the same as in the description of the syntax options,</p> <ul style="list-style-type: none"> <li>• The <b>oid</b> keyword specifies the SNMP object identifier (object ID).</li> <li>• The <i>oid-value</i> argument can be the Object ID value of the data element, in SNMP dotted notation. An OID is defined as a <b>type</b> in the associated MIB, CISCO-EMBEDDED-EVENT-MGR-MIB, and each <b>type</b> has an object value.</li> <li>• The <b>istable</b> keyword specifies whether the OID is an SNMP table.</li> <li>• The <b>sync</b> keyword specifies that the applet is to run in synchronous mode. The return code from the applet indicates whether to reply to the SNMP request. The description for code 0 is “do not reply to the request”</li> </ul> |

|               | Command or Action                                                         | Purpose                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
|---------------|---------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|               |                                                                           | <p>and the description for code 1 is “reply to the request”. When the return code from the applet replies to the request, a value is specified in the applet for the object using <b>action snmp-object-value</b> command.</p> <ul style="list-style-type: none"> <li>• The <b>type</b> keyword specifies the type of object.</li> <li>• The <i>value</i> argument is the value of the object.</li> <li>• The <b>skip</b> keyword specifies whether to skip CLI command execution.</li> <li>• The <b>default</b> keyword specifies the time to process the SET or GET request normally by the applet. If the <b>default</b> keyword is not specified, the default time period is set to 30 seconds.</li> <li>• The <i>milliseconds</i> argument is the time period during which the SNMP Object event detector waits for the policy to exit.</li> <li>• The <b>maxrun</b> keyword specifies the maximum runtime of the applet. If the <b>maxrun</b> keyword is specified, the <i>maxruntime-number</i> value must be specified. If the <b>maxrun</b> keyword is not specified, the default applet run time is 20 seconds.</li> <li>• The <i>milliseconds</i> argument is the maximum runtime of the applet in milliseconds. If the argument is not specified, the default 20-second run-time limit is used.</li> </ul> |
| <b>Step 5</b> | <p><b>exit</b></p> <p><b>Example:</b></p> <pre>Device(config)# exit</pre> | Exits global configuration mode and returns to privileged EXEC mode.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |

## Disabling AAA Authorization

Perform this task to allow EEM policies to bypass AAA authorization when triggered.

### SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **event manager applet** *applet-name* [**authorization bypass**] [**class** *class-options*] [**trap**]
4. **exit**



## DETAILED STEPS

|        | Command or Action                                                                                                                                                                                                                | Purpose                                                                                                            |
|--------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------|
| Step 1 | <b>enable</b><br><b>Example:</b><br>Device> enable                                                                                                                                                                               | Enables privileged EXEC mode. <ul style="list-style-type: none"> <li>• Enter your password if prompted.</li> </ul> |
| Step 2 | <b>configure terminal</b><br><b>Example:</b><br>Device# configure terminal                                                                                                                                                       | Enters global configuration mode.                                                                                  |
| Step 3 | <b>event manager applet</b> <i>applet-name</i> [ <b>authorization bypass</b> ] [ <b>class</b> <i>class-options</i> ] [ <b>trap</b> ]<br><b>Example:</b><br>Device(config)# event manager applet one class A authorization bypass | Registers the applet with the Embedded Event Manager (EEM) and enters applet configuration mode.                   |
| Step 4 | <b>exit</b><br><b>Example:</b><br>Device(config-aaplet)# exit                                                                                                                                                                    | Exits device configuration applet mode and returns to privileged EXEC mode.                                        |

## Configuring Description of an Embedded Event Manager Applet

Perform this task to describe an EEM applet. The description of an applet can be added in any order, before or after any other applet configuration. Configuring a new description for an applet that already has a description overwrites the current description. An applet description is optional.

Perform this task to configure a new description for an applet.

## SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **event manager applet** *applet-name*
4. **description** *line*
5. **event syslog pattern** *regular-expression*
6. **action** *label* **syslog msg** *msg-text*
7. **end**

## DETAILED STEPS

|        | Command or Action                | Purpose                                                                                                            |
|--------|----------------------------------|--------------------------------------------------------------------------------------------------------------------|
| Step 1 | <b>enable</b><br><b>Example:</b> | Enables privileged EXEC mode. <ul style="list-style-type: none"> <li>• Enter your password if prompted.</li> </ul> |

|               | Command or Action                                                                                                                                 | Purpose                                                                                                                                                                                                                                                                                                       |
|---------------|---------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|               | Device> enable                                                                                                                                    |                                                                                                                                                                                                                                                                                                               |
| <b>Step 2</b> | <b>configure terminal</b><br><b>Example:</b><br>Device# configure terminal                                                                        | Enters global configuration mode.                                                                                                                                                                                                                                                                             |
| <b>Step 3</b> | <b>event manager applet</b> <i>applet-name</i><br><b>Example:</b><br>Device(config)# event manager applet increment                               | Registers the applet with the EEM and enters applet configuration mode.                                                                                                                                                                                                                                       |
| <b>Step 4</b> | <b>description</b> <i>line</i><br><b>Example:</b><br>Device(config-applet)# description "This applet looks for the word count in syslog messages" | Adds or modifies the description of an EEM applet that is run by sampling Simple Network Management Protocol (SNMP).                                                                                                                                                                                          |
| <b>Step 5</b> | <b>event syslog pattern</b> <i>regular-expression</i><br><b>Example:</b><br>Device(config-applet)# event syslog pattern "count"                   | Specifies the event criteria for an Embedded Event Manager (EEM) applet that is run by matching syslog messages.                                                                                                                                                                                              |
| <b>Step 6</b> | <b>action</b> <i>label</i> <b>syslog msg</b> <i>msg-text</i><br><b>Example:</b><br>Device(config-applet)# action 1 syslog msg hi                  | Specifies the action to be taken when an EEM applet is triggered. <ul style="list-style-type: none"> <li>• In this example, the action taken is to write a message to syslog.</li> <li>• The <i>msg-text</i> argument can be character text, an environment variable, or a combination of the two.</li> </ul> |
| <b>Step 7</b> | <b>end</b><br><b>Example:</b><br>Device(config-applet)# end                                                                                       | Exits applet configuration mode and returns to privileged EXEC mode.                                                                                                                                                                                                                                          |

# Configuration Examples for Writing Embedded Event Manager Policies Using Tcl

## Embedded Event Manager Applet Configuration Examples

The following examples show how to create an EEM applet for some of the EEM event detectors. These examples follow steps outlined in the [Registering and Defining an Embedded Event Manager Applet, on page 2013](#).

### Application-Specific Event Detector

The following example shows how a policy named EventPublish\_A runs every 20 seconds and publishes an event type numbered 1 to an EEM subsystem numbered 798. The subsystem value of 798 specifies that a publish event has occurred from an EEM policy. A second policy named EventPublish\_B is registered to run when the EEM event type 1 occurs with subsystem 798. When the EventPublish\_B policy runs, it sends a message to syslog containing data passed as an argument from the EventPublish\_A policy.

```
event manager applet EventPublish_A
 event timer watchdog time 20.0
 action 1.0 syslog msg "Applet EventPublish_A"
 action 2.0 publish-event sub-system 798 type 1 arg1 twenty
 exit
event manager applet EventPublish_B
 event application sub-system 798 type 1
 action 1.0 syslog msg "Applet EventPublish_B arg1 $_application_data1"
```

### CLI Event Detector

The following example shows how to specify an EEM applet to run when the Cisco IOS **write memory** CLI command is run. The applet provides a notification that this event has occurred via a syslog message. In the example, the **sync** keyword is configured with the **yes** argument, and this means that the event detector is notified when this policy completes running. The exit status of the policy determines whether the CLI command will be executed. In this example, the policy exit status is set to one and the CLI command runs.

```
event manager applet cli-match
 event cli pattern "write mem.*" sync yes
 action 1.0 syslog msg "$_cli_msg Command Executed"
 set 2.0 _exit_status 1
```

The following example shows an applet which matches the **cli pattern** with the test argument. When **show access-list test** is entered, the CLI event detector matches the test argument, and the applet is triggered. The **debug event manager detector cli** output is added to show num\_matches is set to one.

```
!
event manager applet EEM-PIPE-TEST
 event cli pattern "test" sync yes
 action 1.0 syslog msg "Pattern matched!"
!
*Aug 23 23:19:59.827: check_eem_cli_policy_handler: command_string=show access-lists test
*Aug 23 23:19:59.827: check_eem_cli_policy_handler: num_matches = 1, response_code = 4
*Aug 23 23:19:59.843: %HA_EM-6-LOG: EEM-PIPE-TEST: Pattern matched!
```




---

**Note** The functionality provided in the CLI event detector only allows a regular expression pattern match on a valid IOS CLI command itself. This does not include text after a pipe (|) character when redirection is used.

---

The following example shows that when **show version | include test** is entered, the applet fails to trigger because the CLI event detector does not match on characters entered after the pipe (|) character and the **debug event manager detector cli** output shows num\_matches is set to zero.

```
*Aug 23 23:20:16.827: check_eem_cli_policy_handler: command_string=show version
*Aug 23 23:20:16.827: check_eem_cli_policy_handler: num_matches = 0, response_code = 1
```

### Counter Event Detector and Timer Event Detector

The following example shows that the EventCounter\_A policy is configured to run once a minute and to increment a well-known counter called critical\_errors. A second policy--EventCounter\_B--is registered to be triggered when the well-known counter called critical\_errors exceeds a threshold of 3. When the EventCounter\_B policy runs, it resets the counter to 0.

```
event manager applet EventCounter_A
 event timer watchdog time 60.0
 action 1.0 syslog msg "EventCounter_A"
 action 2.0 counter name critical_errors op inc value 1
 exit
event manager applet EventCounter_B
 event counter name critical_errors entry-op gt entry-val 3 exit-op lt exit-val 3
 action 1.0 syslog msg "EventCounter_B"
 action 2.0 counter name critical_errors op set value 0
```

### Interface Counter Event Detector

The following example shows how a policy named EventInterface is triggered every time the receive\_throttle counter for Fast Ethernet interface 0/0 is incremented by 5. The polling interval to check the counter is specified to run once every 90 seconds.

```
event manager applet EventInterface
 event interface name FastEthernet0/0 parameter receive_throttle entry-op ge entry-val 5
 entry-val-is-increment true poll-interval 90
 action 1.0 syslog msg "Applet EventInterface"
```

### Resource Event Detector

The following example shows how to specify event criteria based on an ERM event report for a policy defined to report high CPU usage:

```
event manager applet policy-one
 event resource policy cpu-high
 action 1.0 syslog msg "CPU high at $_resource_current_value percent"
```

### RF Event Detector

The RF event detector is only available on networking devices that contain dual Route Processors (RPs). The following example shows how to specify event criteria based on an RF state change notification:

```
event manager applet start-rf
 event rf event rf_prog_initialization
 action 1.0 syslog msg "rf state rf_prog_initialization reached"
```

### RPC Event Detector

The RPC event detector allows an outside entity to make a Simple Object Access Protocol (SOAP) request to the device and invokes a defined EEM policy or script. The following example shows how an EEM applet called Event\_RPC is being registered to run an EEM script:

```
event manager applet Event_RPC
 event rpc
 action print puts "hello there"
```

The following example shows the format of the SOAP request and reply message:

```
<?xml version="1.0" encoding="UTF-8"?>
<SOAP:Envelope xmlns:SOAP="http://www.cisco.com/eem.xsd">
 <SOAP:Body>
 <run_eemscript>
 <script_name>Event_RPC</script_name>
 </run_eemscript>
 </SOAP:Body>
</SOAP:Envelope>
]]>]]>
<?xml version="1.0" encoding="UTF-8"?><SOAP:Envelope
xmlns:SOAP="http://www.cisco.com/eem.xsd"><SOAP:Body>
<run_eemscript_response><return_code>0</return_code><output></output></run_eemscript_response></SOAP:Body></SOAP:Envelope>]]>]]>
```

### SNMP Event Detector

The following example shows how to specify an EEM applet to run when the CPU usage is greater than 75 percent. When the EEM applet runs, the CLI commands **enable** and **show cpu processes** are run, and an e-mail containing the result of the **show cpu processes** command is sent to an engineer.

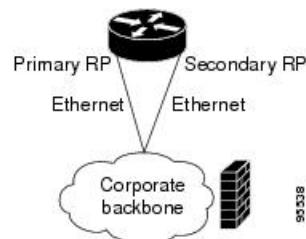
```
event manager applet snmpcpu75
 event snmp oid 1.3.6.1.4.1.9.9.109.1.1.1.1.3.1 get-type exact entry-op ge entry-val 75
 poll-interval 10
 action 1.0 cli command "enable"
 action 2.0 cli command "show process cpu"
 action 3.0 mail server "192.168.1.146" to "engineer@cisco.com" from "devtest@cisco.com"
 subject "B25 PBX Alert" body "$_cli_result"
```

The next example is more complex and shows how to configure an EEM applet that causes a switch to the secondary (redundant) Route Processor (RP) when the primary RP runs low on memory.

This example illustrates a method for taking preventative action against a software fault that causes a memory leak. The action taken here is designed to reduce downtime by switching over to a redundant RP when a possible memory leak is detected.

The figure below shows a dual RP device that is running an EEM image. An EEM applet has been registered through the CLI using the **event manager applet** command. The applet will run when the available memory on the primary RP falls below the specified threshold of 5,120,000 bytes. The applet actions are to write a message to syslog that indicates the number of bytes of memory available and to switch to the secondary RP.

**Figure 141: Dual RP Topology**



The commands used to register the policy are shown below.

```
event manager applet memory-demo
 event snmp oid 1.3.6.1.4.1.9.9.48.1.1.1.6.1 get-type exact entry-op lt entry-val 5120000
 poll-interval 90
 action 1.0 syslog priority critical msg "Memory exhausted; current available memory is
```

```
$_snmp_oid_val bytes"
action 2.0 force-switchover
```

The registered applet is displayed using the **show event manager policy registered** command:

```
Device# show event manager policy registered
No. Type Event Type Time Registered Name
1 applet snmp Thu Jan30 05:57:16 2003 memory-demo
oid {1.3.6.1.4.1.9.9.48.1.1.1.6.1} get-type exact entry-op lt entry-val {5120000}
poll-interval 90
action 1.0 syslog priority critical msg "Memory exhausted; current available memory is
$_snmp_oid_val bytes"
action 2.0 force-switchover
```

For the purpose of this example, a memory depletion is forced on the device, and a series of **show memory** commands are executed to watch the memory deplete:

```
Device# show memory
 Head Total(b) Used(b) Free(b) Lowest(b) Largest(b)
Processor 53585260 212348444 119523060 92825384 92825384 92365916
Fast 53565260 131080 70360 60720 60720 60668
Device# show memory
 Head Total(b) Used(b) Free(b) Lowest(b) Largest(b)
Processor 53585260 212364664 164509492 47855172 47855172 47169340
Fast 53565260 131080 70360 60720 60720 60668
Device# show memory
 Head Total(b) Used(b) Free(b) Lowest(b) Largest(b)
Processor 53585260 212369492 179488300 32881192 32881192 32127556
Fast 53565260 131080 70360 60720 60720 60668
```

When the threshold is reached, an EEM event is triggered. The applet named memory-demo runs, causing a syslog message to be written to the console and a switch to be made to the secondary RP. The following messages are logged:

```
00:08:31: %HA_EM-2-LOG: memory-demo: Memory exhausted; current available memory is
4484196 bytes
00:08:31: %HA_EM-6-FMS_SWITCH_HARDWARE: fh_io_msg: Policy has requested a hardware
switchover
```

The following is partial output from the **show running-config** command on both the primary RP and the secondary (redundant) RP:

```
redundancy
mode sso
.
!
event manager applet memory-demo
event snmp oid 1.3.6.1.4.1.9.9.48.1.1.1.6.1 get-type exact entry-op lt entry-val
5120000 poll-interval 90
action 1.0 syslog priority critical msg "Memory exhausted; current available memory
is $_snmp_oid_val bytes"
action 2.0 force-switchover
```

### SNMP Notification Event Detector

The following example shows how to configure the **snmp-server community public RW** and **snmp-server manager** commands before **event snmp-notification** is configured.

```
snmp-server community public RW
snmp-server manager
```

The following example shows how an EEM applet called SNMP\_Notification is being registered to run an EEM script when the device receives an SNMP notification on destination IP address 192.168.1.1 for object ID 1 whose value equals 10.

```
event manager applet SNMP_Notification
 event snmp-notification dest_ip_address 192.168.1.1 oid 1 op eq oid-value 10
 action 1 policy eem_script
```

### Syslog Event Detector

The following example shows how to specify an EEM applet to run when syslog identifies that Ethernet interface 1/0 is down. The applet sends a message about the interface to syslog.

```
event manager applet interface-down
 event syslog pattern \.*UPDOWN.*Ethernet1/0.*" occurs 4
 action 1.0 syslog msg "Ethernet interface 1/0 changed state 4 times"
```

## Configuration Examples for Embedded Event Manager Applet

### Example Identity Event Detector

The following example shows how a policy named “EventIdentity” is triggered every time the authentication on the Fast Ethernet interface 0 is success.

```
event manager applet EventIdentity
 event identity interface FastEthernet0 authc success
 action 1.0 syslog msg "Applet EventIdentity"
```

### Example MAT Event Detector

The following example shows how a policy named “EventMat” is triggered every time a mac-address is learned in the mac-address-table.

```
event manager applet EventMat
 event mat interface FastEthernet0
 action 1.0 syslog msg "Applet EventMat"
```

### Example Neighbor-Discovery Event Detector

The following example shows how a policy named “EventNeighbor” is triggered when a Cisco Discovery Protocol (CDP) cache entry changes.

```
event manager applet EventNeighbor
 event neighbor-discovery interface FastEthernet0 cdp all
 action 1.0 syslog msg "Applet EventNeighbor"
```

## Embedded Event Manager Manual Policy Execution Examples

The following examples show how to use the none event detector to configure an EEM policy (applet or script) to be run manually.

### Using the event manager run Command

This example shows how to run a policy manually using the **event manager run** command. The policy is registered using the **event none** command under applet configuration mode and then run from global configuration mode using the **event manager run** command.

```
event manager applet manual-policy
 event none
 action 1.0 syslog msg "Manual-policy triggered"
end
!
event manager run manual-policy
```

### Using the action policy Command

This example shows how to run a policy manually using the **action policy** command. The policy is registered using the **event none** command under applet configuration mode, and then the policy is executed using the **action policy** command in applet configuration mode.

```
event manager applet manual-policy
 event none
 action 1.0 syslog msg "Manual-policy triggered"
 exit
!
event manager applet manual-policy-two
 event none
 action 1.0 policy manual-policy
 end
!
event manager run manual-policy-two
```

## Embedded Event Manager Watchdog System Monitor (Cisco IOS) Event Detector Configuration Example

The following example shows how to configure three EEM applets to demonstrate how the Cisco IOS watchdog system monitor (IOSWDSysMon) event detector works.

### Watchdog System Monitor Sample1 Policy

The first policy triggers an applet when the average CPU usage for the process named IP Input is greater than or equal to 1 percent for 10 seconds:

```
event manager applet IOSWD_Sample1
 event ioswdsysmon sub1 cpu-proc taskname "IP Input" op ge val 1 period 10
 action 1.0 syslog msg "IOSWD_Sample1 Policy Triggered"
```



### Watchdog System Monitor Sample2 Policy

The second policy triggers an applet when the total amount of memory used by the process named Net Input is greater than 100 kb:

```
event manager applet IOSWD_Sample2
 event ioswdsysmon sub1 mem-proc taskname "Net Input" op gt val 100 is-percent false
 action 1.0 syslog msg "IOSWD_Sample2 Policy Triggered"
```

### Watchdog System Monitor Sample3 Policy

The third policy triggers an applet when the total amount of memory used by the process named IP RIB Update has increased by more than 50 percent over the sample period of 60 seconds:

```
event manager applet IOSWD_Sample3
 event ioswdsysmon sub1 mem-proc taskname "IP RIB Update" op gt val 50 is-percent true
 period 60
 action 1.0 syslog msg "IOSWD_Sample3 Policy Triggered"
```

The three policies are configured, and then repetitive large pings are made to the networking device from several workstations, causing the networking device to register some usage. This will trigger policies 1 and 2, and the console will display the following messages:

```
00:42:23: %HA_EM-6-LOG: IOSWD_Sample1: IOSWD_Sample1 Policy Triggered
00:42:47: %HA_EM-6-LOG: IOSWD_Sample2: IOSWD_Sample2 Policy Triggered
```

To view the policies that are registered, use the **show event manager policy registered** command:

```
Device# show event manager policy registered
No. Class Type Event Type Trap Time Registered Name
1 applet system ioswdsysmon Off Fri Jul 23 02:27:28 2004 IOSWD_Sample1
 sub1 cpu_util {taskname {IP Input} op ge val 1 period 10.000 }
 action 1.0 syslog msg "IOSWD_Sample1 Policy Triggered"
2 applet system ioswdsysmon Off Fri Jul 23 02:23:52 2004 IOSWD_Sample2
 sub1 mem_used {taskname {Net Input} op gt val 100 is_percent FALSE}
 action 1.0 syslog msg "IOSWD_Sample2 Policy Triggered"
3 applet system ioswdsysmon Off Fri Jul 23 03:07:38 2004 IOSWD_Sample3
 sub1 mem_used {taskname {IP RIB Update} op gt val 50 is_percent TRUE period 60.000 }
 action 1.0 syslog msg "IOSWD_Sample3 Policy Triggered"
```

## Configuration SNMP Library Extensions Examples

### SNMP Get Operations Examples

The following example shows how to send a get request to the local host.

```
Device(config)# event manager applet snmp
Device(config-applet)# event snmp oid
 1.3.6.1.2.1.1.1.0 get-type exact entry-op
 lt entry-val
 5120000 poll-interval
 90
Device(config-applet)# action 1.3 info type snmp oid
 1.3.6.1.2.1.1.1.0 get-type exact
 community
 public
Device(config-applet)# action 1.3 info type snmp oid
```

```
1.3.6.1.2.1.1.4.0 get-type next community
public
```

The following log message will be written to the SNMP event manager log:

```
1d03h:%HA_EM-6-LOG: lg: 1.3.6.1.2.1.1.1.0
1d04h:%HA_EM-6-LOG: lgn: 1.3.6.1.2.1.1.5.0
```

The following example shows how to send a get request to a remote host.

```
Device(config)# event manager applet snmp
Device(config-applet)# event snmp oid
1.3.6.1.2.1.1.1.0 get-type exact entry-op
lt entry-val
5120000 poll-interval
90
Device(config-applet)# action 1.3 info type snmp oid
1.3.6.1.2.1.1.4.0 get-type next community
public ipaddr
172.17.16.69
Device(config-applet)# action 1.3 info type snmp getid
1.3.6.1.2.1.1.1.0 community
public ipaddr
172.17.16.69
```

The following log message is written to the SNMP event manager log:

```
1d03h:%HA_EM-6-LOG: lg: 1.3.6.1.2.1.1.1.0
1d04h:%HA_EM-6-LOG: lgn: 1.3.6.1.2.1.1.5.0
```

## SNMP GetID Operations Examples

The following example shows how to send a getid request to the local host.

```
Device(config)# event manager applet snmp
Device(config-applet)# event snmp oid
1.3.6.1.2.1.1.1.0 get-type exact entry-op
lt entry-val
5120000 poll-interval
90
Device(config-applet)# action 1.3 info type snmp getid
community
public
```

The following log message is written to the SNMP event manager log:

```
1d04h:%HA_EM-6-LOG: lgid: _info_snmp_sysname_oid=1.3.6.1.2.1.1.5.0
1d04h:%HA_EM-6-LOG: lgid: _info_snmp_sysname_value=jubjub.cisco.com
1d04h:%HA_EM-6-LOG: lgid: _info_snmp_syslocation_oid=1.3.6.1.2.1.1.6.0
1d04h:%HA_EM-6-LOG: lgid: _info_snmp_syslocation_value=
1d04h:%HA_EM-6-LOG: lgid: _info_snmp_sysdescr_oid=1.3.6.1.2.1.1.1.0
1d04h:%HA_EM-6-LOG: lgid: _info_snmp_sysobjectid_oid=1.3.6.1.2.1.1.2.0
1d04h:%HA_EM-6-LOG: lgid: _info_snmp_sysobjectid_value=products.222
1d04h:%HA_EM-6-LOG: lgid: _info_snmp_sysuptime_oid=1.3.6.1.2.1.1.3.0
1d04h:%HA_EM-6-LOG: lgid: _info_snmp_sysuptime_oid=10131676
1d04h:%HA_EM-6-LOG: lgid: _info_snmp_syscontact_oid=1.3.6.1.2.1.1.4.0
1d04h:%HA_EM-6-LOG: lgid: _info_snmp_syscontact_value=YYY
```

The following example shows how to send a getid request to a remote host.

```

Device(config)# event manager applet snmp
Device(config-applet)# event snmp oid
1.3.6.1.2.1.1.1.0 get-type exact entry-op
lt entry-val
5120000 poll-interval
90
Device(config-applet)# action 1.3 info type snmp getid
1.3.6.1.2.1.1.1.0 community
public ipaddr
172.17.16.69

```

The following log message is written to the SNMP event manager log:

```

1d04h:%HA_EM-6-LOG: lgid: _info_snmp_sysname_oid=1.3.6.1.2.1.1.5.0
1d04h:%HA_EM-6-LOG: lgid: _info_snmp_sysname_value=jubjub.cisco.com
1d04h:%HA_EM-6-LOG: lgid: _info_snmp_syslocation_oid=1.3.6.1.2.1.1.6.0
1d04h:%HA_EM-6-LOG: lgid: _info_snmp_syslocation_value=
1d04h:%HA_EM-6-LOG: lgid: _info_snmp_sysdescr_oid=1.3.6.1.2.1.1.1.0
1d04h:%HA_EM-6-LOG: lgid: _info_snmp_sysobjectid_oid=1.3.6.1.2.1.1.2.0
1d04h:%HA_EM-6-LOG: lgid: _info_snmp_sysobjectid_value=products.222
1d04h:%HA_EM-6-LOG: lgid: _info_snmp_sysuptime_oid=1.3.6.1.2.1.1.3.0
1d04h:%HA_EM-6-LOG: lgid: _info_snmp_sysuptime_oid=10131676
1d04h:%HA_EM-6-LOG: lgid: _info_snmp_syscontact_oid=1.3.6.1.2.1.1.4.0
1d04h:%HA_EM-6-LOG: lgid: _info_snmp_syscontact_value=YYY

```

## Set Operations Examples

The following example shows how to perform a set operation on the local host.

```

Device(config)# event manager applet snmp
Device(config-applet)# event snmp oid
1.3.6.1.2.1.1.1.0 get-type exact entry-op
lt entry-val
5120000 poll-interval
90
Device(config-applet)# action 1.3 info type snmp oid
1.3.6.1.2.1.1.4.0 set-type
integer
5 sysName.0 community
public

```

The following log message is written to the SNMP event manager log:

```

1d04h:%HA_EM-6-LOG: lset: 1.3.6.1.2.1.1.4.0
1d04h:%HA_EM-6-LOG: lset: XXX

```

The following example shows how to perform a set operation on a remote host.

```

Device(config)# event manager applet snmp
Device(config-applet)# event snmp oid
1.3.6.1.2.1.1.1.0 get-type exact entry-op
lt entry-val
5120000 poll-interval
90
Device(config-applet)# action 1.3 info type snmp oid
1.3.6.1.2.1.1.4.0 set-type integer
5 sysName.0 community
public ipaddr
172.17.16.69

```

The following log message is written to the SNMP event manager log:

```
1d04h:%HA_EM-6-LOG: lset: 1.3.6.1.2.1.1.4.0
1d04h:%HA_EM-6-LOG: lset: XXX
```

## Generating SNMP Notifications Examples

The following example shows how to configure SNMP traps for the sysUpTime.0 variable:

```
Device(config)# event manager applet snmp
Device(config-applet)# event snmp oid
1.3.6.1.4.1.9.9.48.1.1.1.6.1 get-type exact entry-op
lt entry-val
5120000 poll-interval
90
Device(config-applet)# action 1.3 info type snmp var
sysUpTime.0 oid
1.3.6.1.4.1.9.9.43.1.1.6.1.3.41 integer
2
Device(config-applet)# action 1.4 info type snmp trap
enterprise-oid
ciscoSyslogMIB.2 generic-trapnum
6 specific-trapnum
1 trap-oid
1.3.6.1.4.1.9.9.41.2.0.1 trap-var
sysUpTime.0
```

The following output is generated if the debug snmp packets command is enabled:

```
Device# debug snmp packets
1d04h: SNMP: Queuing packet to 172.69.16.2
1d04h: SNMP: V1 Trap, ent ciscoSyslogMIB.2, addr 172.19.rap 1
clogHistoryEntry.3 = 4
clogHistoryEntry.6 = 9999
1d04h: SNMP: Queuing packet to 172.19.208.130
1d04h: SNMP: V1 Trap, ent ciscoSyslogMIB.2, addr 172.19.rap 1
clogHistoryEntry.3 = 4
clogHistoryEntry.6 = 9999
1d04h: SNMP: Packet sent via UDP to 172.69.16.2
1d04h: SNMP: Packet sent via UDP to 172.69.16.2
infra-viewl0:
Packet Dump:
30 53 02 01 00 04 04 63 6f 6d 6d a4 48 06 09 2b
06 01 04 01 09 09 29 02 40 04 ac 13 d1 17 02 01
06 02 01 01 43 04 00 9b 82 5d 30 29 30 12 06 0d
2b 06 01 04 01 09 09 29 01 02 03 01 03 02 01 04
30 13 06 0d 2b 06 01 04 01 09 09 29 01 02 03 01
06 02 02 27 0f
Received SNMPv1 Trap:
Community: comm
Enterprise: ciscoSyslogMIBNotificationPrefix
Agent-addr: 172.19.209.23
Enterprise Specific trap.
Enterprise Specific trap: 1
Time Ticks: 10191453
clogHistSeverity = error(4)
clogHistTimestamp = 9999
```

The following example shows how to configure SNMP inform requests for the sysUpTime.0 variable:

```

Device(config)# event manager applet snmp
Device(config-applet)# event snmp oid
1.3.6.1.4.1.9.9.48.1.1.1.6.1 get-type exact entry-op
 lt entry-val
 5120000 poll-interval
 90
Device(config-applet)# action 1.3 info type snmp var
 sysUpTime.0 oid
1.3.6.1.4.1.9.9.43.1.1.6.1.3.41 integer
 2
Device(config-applet)# action 1.4 info type snmp inform
 trap-oid
1.3.6.1.4.1.9.9.43.2.0.1 trap-var
 sysUpTime.0 community
 public ipaddr
 172.19.209.24

```

The following output is generated if the debug snmp packets command is enabled:

```

Device# debug snmp packets
1d04h: SNMP: Inform request, reqid 24, errstat 0, erridx 0
sysUpTime.0 = 10244391
snmpTrapOID.0 = ciscoConfigManMIB.2.0.1
ccmHistoryEventEntry.3.40 = 1
1d04h: SNMP: Packet sent via UDP to 172.19.209.24.162
1d04h: SNMP: Packet received via UDP from 172.19.209.24 on FastEthernet0/0
1d04h: SNMP: Response, reqid 24, errstat 0, erridx 0
1d04h: SNMP: Response, reqid 24, errstat 0, erridx 0
1d04h: SNMP: Inform request, reqid 25, errstat 0, erridx 0
sysUpTime.0 = 10244396
snmpTrapOID.0 = ciscoConfigManMIB.2.0.1
ccmHistoryEventEntry.3.41 = 2
1d04h: SNMP: Packet sent via UDP to 172.19.209.24.162
1d04h: SNMP: Packet received via UDP from 172.19.209.24 on FastEthernet0/0
1d04h: SNMP: Response, reqid 25, errstat 0, erridx 0
1d04h: SNMP: Response, reqid 25, errstat 0, erridx 0
Device# debug snmp packets
5d04h: SNMP: Packet received via UDP from 172.19.209.23 on FastEthernet0/0
5d04h: SNMP: Inform request, reqid 24, errstat 0, erridx 0
sysUpTime.0 = 10244391
snmpTrapOID.0 = ciscoConfigManMIB.2.0.1
ccmHistoryEventEntry.3.40 = 1
5d04h: dest if_index = 1
5d04h: dest ip addr= 172.19.209.24
5d04h: SNMP: Response, reqid 24, errstat 0, erridx 0
5d04h: SNMP: Packet sent via UDP to 172.19.209.23.57748
5d04h: SNMP: Packet received via UDP from 172.19.209.23 on FastEthernet0/0
5d04h: SNMP: Inform request, reqid 25, errstat 0, erridx 0

```

## Configuring Variable Logic for EEM Applets Examples

The following sections provide examples on some selected action commands. For information on all the action commands supporting variable logic within applets, see the table below.

In this example, conditional loops **while**, **if** and **foreach** are used to print data. Other action commands such as **action divide**, **action increment** and **action puts** are used to define the actions to be performed when the conditions are met.

```
event manager applet printdata
```

```

event none
action 100 set colors "red green blue"
action 101 set shapes "square triangle rectange"
action 102 set i "1"
action 103 while $i lt 6
action 104 divide $i 2
action 105 if $_remainder eq 1
action 106 foreach_iterator "$colors"
action 107 puts newline "$_iterator "
action 108 end
action 109 puts ""
action 110 else
action 111 foreach_iterator "$shapes"
action 112 puts newline "$_iterator "
action 113 end
action 114 puts ""
action 115 end
action 116 increment i
action 117 end

```

When the event manager applet ex is run, the following output is obtained:

```

event manager run printdata
red green blue
square triangle rectange
red green blue
square triangle rectange
red green blue

```

In this example, two environment variables `poll_interface` and `max_rx_rate` are set to `F0/0` and `3` respectively. Every 30 seconds there is a poll on an interface for rx rate. If the rx rate is greater than the threshold, a syslog message is displayed.

This applet makes use of the `foreach` conditional statement to poll the interface, the `if` conditional block to compare the value under `RXPS` with `max_rx_rate` that was set in the EEM environment variable.

```

event manager environment poll_interfaces F0/0
event manager environment max_rx_rate 3
ev man app check_rx_rate
ev timer watchdog name rx_timer time 30
action 100 foreach int $poll_interfaces
action 101 cli command "en"
action 102 cli command "show int $int summ | beg -----"
action 103 foreach line $_cli_result "\n"
action 105 regexp ".*[0-9]+\s+[0-9]+\s+[0-9]+\s+[0-9]+\s+[0-9]+\s+[0-9]+\s+([0-9])\s+.*" $line
junk rxps
action 106 if $_regexp_result eq 1
action 107 if $rxps gt $max_rx_rate
action 108 syslog msg "Warning rx rate for $int is > than threshold. Current value is
$rxps
(threshold is $max_rx_rate)"
action 109 end
action 110 end
action 111 end
action 112 end

```

Example syslog message:

```

Oct 16 09:29:26.153: %HA_EM-6-LOG: c: Warning rx rate for F0/0 is > than threshold.
Current value is 4 (threshold is 3)
The output of show int F0/0 summ is of the format:

```

```
#show int f0/0 summ

*: interface is up
IHQ: pkts in input hold queue IQD: pkts dropped from input queue
OHQ: pkts in output hold queue OQD: pkts dropped from output queue
RXBS: rx rate (bits/sec) RXPS: rx rate (pkts/sec)
TXBS: tx rate (bits/sec) TXPS: tx rate (pkts/sec)
TRTL: throttle count

Interface IHQ IQD OHQ OQD RXBS RXPS TXBS TXPS TRTL

* FastEthernet0/0 0 87283 0 0 0 0 0 0 0
```



**Note** To use other action commands supporting variable logic within applets, use the commands listed in the table below.

**Table 202: Available action commands**

| Action Commands         | Purpose                                                                                                                                  |
|-------------------------|------------------------------------------------------------------------------------------------------------------------------------------|
| action add              | Adds the value of two variables when an EEM applet is triggered.                                                                         |
| action append           | Appends the given value to the current value of a variable when an EEM applet is triggered.                                              |
| action break            | Causes an immediate exit from a loop of actions when an EEM applet is triggered                                                          |
| action comment          | Adds comments to an applet when an EEM applet is triggered                                                                               |
| action context retrieve | Retrieves variables identified by a given set of context name keys when an EEM applet is triggered.                                      |
| action context save     | Saves information across multiple policy triggers when an EEM applet is triggered.                                                       |
| action continue         | Continues with a loop of actions when an EEM applet is triggered.                                                                        |
| action decrement        | Decrements the value of a variable when an EEM applet is triggered.                                                                      |
| action divide           | Divides the dividend value by the given divisor value when an EEM applet is triggered.                                                   |
| action else             | Specifies the beginning of else conditional action block in if / else conditional action block when an EEM applet is triggered.          |
| action elseif           | Identifies the beginning of the else conditional action block in the else / if conditional action block when an EEM applet is triggered. |

| Action Commands                  | Purpose                                                                                                                                                     |
|----------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------|
| action end                       | Specifies the identification of the end of an conditional action block in the if / else and while conditional action block when an EEM applet is triggered. |
| action exit                      | Specifies an immediate exit from the running applet configuration when an EEM applet is triggered.                                                          |
| action foreach                   | Specifies the iteration of an input string using the delimiter as a tokenizing pattern, when an EEM applet is triggered.                                    |
| action gets                      | Gets an input from the local TTY in a synchronous applet and store the value in the given variable when an EEM applet is triggered.                         |
| action if                        | Specifies the identification of the beginning of an if conditional block when an EEM applet is triggered.                                                   |
| action if goto                   | Instructs the applet to jump to a given label if the specified condition is true when an EEM applet is triggered.                                           |
| action increment                 | Increments the value of a variable when an EEM applet is triggered.                                                                                         |
| action info type interface-names | Specifies the action of obtaining interface names when an EEM applet is triggered.                                                                          |
| action info type snmp getid      | Retrieves the individual variables from a Simple Network Management Protocol (SNMP) entity during the SNMP get operation.                                   |
| action info type snmp inform     | Sends an SNMP inform requests when an EEM applet is triggered.                                                                                              |
| action info type snmp oid        | Specifies the type of SNMP get operation and the object to retrieve during the SNMP set operation, when an EEM applet is triggered.                         |
| action info type snmp trap       | Sends SNMP trap requests when an EEM applet is triggered.                                                                                                   |
| action info type snmp var        | Creates a variable for an SNMP object identifier (OID) and its value from an EEM applet                                                                     |
| action multiply                  | Specifies the action of multiplying the variable value with a specified given integer value when an EEM applet is triggered.                                |
| action puts                      | Enables the action of printing data directly to the local tty when an EEM applet is triggered.                                                              |



| Action Commands       | Purpose                                                                                                                                       |
|-----------------------|-----------------------------------------------------------------------------------------------------------------------------------------------|
| action regexp         | Specifies the action of matching a regular expression pattern on an input string when an EEM applet is triggered.                             |
| action set (EEM)      | Specifies the action of setting the value of a variable when an EEM applet is triggered.                                                      |
| action string compare | Specifies the action of comparing two unequal strings when an EEM applet is triggered.                                                        |
| action string equal   | Specifies the action of verifying whether or not two strings are equal when an EEM applet is triggered                                        |
| action string first   | Specifies the action of returning the index on the first occurrence of string1 within string2 when an EEM applet is triggered.                |
| action string index   | Specifies the action of returning the characters specified at a given index value when an EEM applet is triggered.                            |
| action string last    | Specifies the action of returning the index on the last occurrence of string1 within string 2 when an EEM applet is triggered.                |
| action string length  | Specifies the action of returning the number of characters in a string when the EEM applet is triggered.                                      |
| action string match   | Specifies the action of returning 1 to the <code>\$_string_result</code> , if the string matches the pattern when an EEM applet is triggered. |
| action string range   | Specifies the action of storing a range of characters in a string when an EEM applet is triggered.                                            |
| action string replace | Specifies the action of storing a new string by replacing range of characters in the specified string when an EEM applet is triggered.        |
| action string tolower | Specifies the action of storing specific range of characters of a string in lowercase when an EEM applet is triggered.                        |
| action string toupper | Specifies the action of storing specific range of characters of a string in uppercase when an EEM applet is triggered.                        |
| action string trim    | Specifies the action to trim a string when an EEM applet is triggered.                                                                        |

| Action Commands         | Purpose                                                                                                                        |
|-------------------------|--------------------------------------------------------------------------------------------------------------------------------|
| action string trimleft  | Specifies the action to trim the characters of one string from the left end of another string when an EEM applet is triggered. |
| action string trimright | Specifies the action to trim the characters one string from the right end of another string when an EEM applet is triggered.   |
| action subtract         | Subtracts the value of a variable from another value when an EEM applet is triggered.                                          |
| action while            | Specifies the action of identifying the beginning of a loop of conditional block when an EEM applet is triggered.              |

## Configuring Event SNMP-Object Examples

The following example shows the SET operation and the value to set is in `$_snmp_value` and it is managed by the script. The example below saves the oid and its value as contexts to be retrieved later.

```
event manager applet snmp-object1
 description "APPLET SNMP-OBJ-1"
 event snmp-object oid 1.3.6.1.2.1.31.1.1.1.18 type string sync no skip no istable yes
 default 0
 action 1 syslog msg "SNMP-OBJ1:TRIGGERED" facility "SNMP_OBJ"
 action 2 context save key myoid variable "_snmp_oid"
 action 3 context save key myvalue variable "_snmp_value"
```

## Configuring Description of an EEM Applet Examples

The following example shows how to add or modify the description for an Embedded Event Manager (EEM) applet that is run by sampling Simple Network Management Protocol (SNMP):

```
event manager applet test
 description "This applet looks for the word count in syslog messages"
 event syslog pattern "count"
 action 1 syslog msg hi
```

## Additional References

The following sections provide references related to writing EEM policies Using the Cisco IOS CLI.

**Related Documents**

| Related Topic                                                                                                  | Document Title                                                     |
|----------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------|
| EEM commands: complete command syntax, defaults, command mode, command history, usage guidelines, and examples | <a href="#">Cisco IOS Embedded Event Manager Command Reference</a> |
| Embedded Event Manager overview                                                                                | Embedded Event Manager Overview module                             |
| Embedded Event Manager policy writing using Tcl                                                                | Writing Embedded Event Manager Policies Using Tcl module           |
| Configuring enhanced object tracking                                                                           | Configuring Enhanced Object Tracking module                        |

**Standards**

| Standard                                                                                              | Title |
|-------------------------------------------------------------------------------------------------------|-------|
| No new or modified standards are supported, and support for existing standards has not been modified. | --    |

**MIBs**

| MIB                          | MIBs Link                                                                                                                                                                                                              |
|------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| CISCO-EMBEDDED-EVENT-MGR-MIB | To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL:<br><a href="http://www.cisco.com/go/mibs">http://www.cisco.com/go/mibs</a> |

**RFCs**

| RFC                                                                                         | Title |
|---------------------------------------------------------------------------------------------|-------|
| No new or modified RFCs are supported, and support for existing RFCs has not been modified. | --    |

**Technical Assistance**

| Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             | Link                                                                                                                     |
|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------|
| <p>The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies.</p> <p>To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds.</p> <p>Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.</p> | <p><a href="http://www.cisco.com/cisco/web/support/index.html">http://www.cisco.com/cisco/web/support/index.html</a></p> |

## Feature Information for Writing EEM 4.0 Policies Using the Cisco IOS CLI

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to [www.cisco.com/go/cfn](http://www.cisco.com/go/cfn). An account on Cisco.com is not required.

**Table 203: Feature Information for Writing EEM 4.0 Policies Using the Cisco IOS CLI**

| Feature Name               | Releases  | Feature Information                                                    |
|----------------------------|-----------|------------------------------------------------------------------------|
| Embedded Event Manager 4.0 | 15.2(5)E1 | This feature was introduced and is supported only on c2960cx platform. |



## CHAPTER 101

# Writing Embedded Event Manager Policies Using Tcl

---

This module describes how software developers can write and customize Embedded Event Manager (EEM) policies using Tool command language (Tcl) scripts to handle Cisco software faults and events. EEM is a policy-driven process by means of which faults in the Cisco software system are reported through a defined application programming interface (API). The EEM policy engine receives notifications when faults and other events occur. EEM policies implement recovery on the basis of the current state of the system and the actions specified in the policy for a given event. Recovery actions are triggered when the policy is run.

- [Prerequisites for Writing Embedded Event Manager Policies Using Tcl, on page 2075](#)
- [Information About Writing Embedded Event Manager Policies Using Tcl, on page 2075](#)
- [How to Write Embedded Event Manager Policies Using Tcl, on page 2082](#)
- [Configuration Examples for Writing Embedded Event Manager Policies Using Tcl, on page 2111](#)
- [Additional References, on page 2132](#)

## Prerequisites for Writing Embedded Event Manager Policies Using Tcl

- Before writing EEM policies, you should be familiar with the “ Embedded Event Manager Overview ” module.
- If you want to write EEM policies using the command-line interface (CLI) commands, you should be familiar with the “ Writing Embedded Event Manager Policies Using the Cisco IOS CLI ” module.

## Information About Writing Embedded Event Manager Policies Using Tcl

### EEM Policies

EEM offers the ability to monitor events and take informational or corrective action when the monitored events occur or reach a threshold. An EEM policy is an entity that defines an event and the actions to be taken

when that event occurs. There are two types of EEM policies: an applet or a script. An applet is a simple form of policy that is defined within the command-line interface (CLI) configuration. A script is a form of policy that is written in Tool Command Language (Tcl).

### EEM Applet

An EEM applet is a concise method for defining event screening criteria and the actions to be taken when that event occurs. In EEM applet configuration mode, three types of configuration statements are supported. The event commands are used to specify the event criteria to trigger the applet to run, the action commands are used to specify an action to perform when the EEM applet is triggered, and the **set** command is used to set the value of an EEM applet variable. Currently only the `_exit_status` variable is supported for the **set** command.

Only one event configuration command is allowed within an applet configuration. When applet configuration submode is exited and no event command is present, a warning is displayed stating that no event is associated with the applet. If no event is specified, the applet is not considered registered. When no action is associated with the applet, events are still triggered but no actions are performed. Multiple action configuration commands are allowed within an applet configuration. Use the **show event manager policy registered** command to display a list of registered applets.

Before modifying an EEM applet, be aware that the existing applet is not replaced until you exit applet configuration mode. While you are in applet configuration mode modifying the applet, the existing applet may be executing. It is safe to modify the applet without unregistering it, because changes are written to a temporary file. When you exit applet configuration mode, the old applet is unregistered and the new version is registered.

Action configuration commands within an applet are uniquely identified using the *label* argument, which can be any string value. Actions are sorted within an applet in ascending alphanumeric key sequence using the *label* argument as the sort key, and they are run using this sequence. The same *label* argument can be used in different applets; the labels must be unique only within one applet.

The Embedded Event Manager schedules and runs policies on the basis of an event specification that is contained within the policy itself. When applet configuration mode is exited, EEM examines the event and action commands that are entered and registers the applet to be run when a specified event occurs.

For more details about writing EEM policies using the Cisco IOS CLI, see the “Writing Embedded Event Manager Policies Using the Cisco IOS CLI” module.

### EEM Script

All Embedded Event Manager scripts are written in Tcl. Tcl is a string-based command language that is interpreted at run time. The version of Tcl supported is Tcl version 8.3.4 plus added script support. Scripts are defined using an ASCII editor on another device, not on the networking device. The script is then copied to the networking device and registered with EEM. Tcl scripts are supported by EEM. As an enforced rule, Embedded Event Manager policies are short-lived run time routines that must be interpreted and executed in less than 20 seconds of elapsed time. If more than 20 seconds of elapsed time are required, the `maxrun` parameter may be specified in the `event_register` statement to specify any desired value.

EEM policies use the full range of the Tcl language’s capabilities. However, Cisco provides enhancements to the Tcl language in the form of Tcl command extensions that facilitate the writing of EEM policies. The main categories of Tcl command extensions identify the detected event, the subsequent action, utility information, counter values, and system information.

EEM allows you to write and implement your own policies using Tcl. Writing an EEM script involves:

- Selecting the event Tcl command extension that establishes the criteria used to determine when the policy is run.
- Defining the event detector options associated with detecting the event.
- Choosing the actions to implement recovery or respond to the detected event.

## EEM Policy Tcl Command Extension Categories

There are different categories of EEM policy Tcl command extensions.



**Note** The Tcl command extensions available in each of these categories for use in all EEM policies are described in later sections in this document.

*Table 204: EEM Policy Tcl Command Extension Categories*

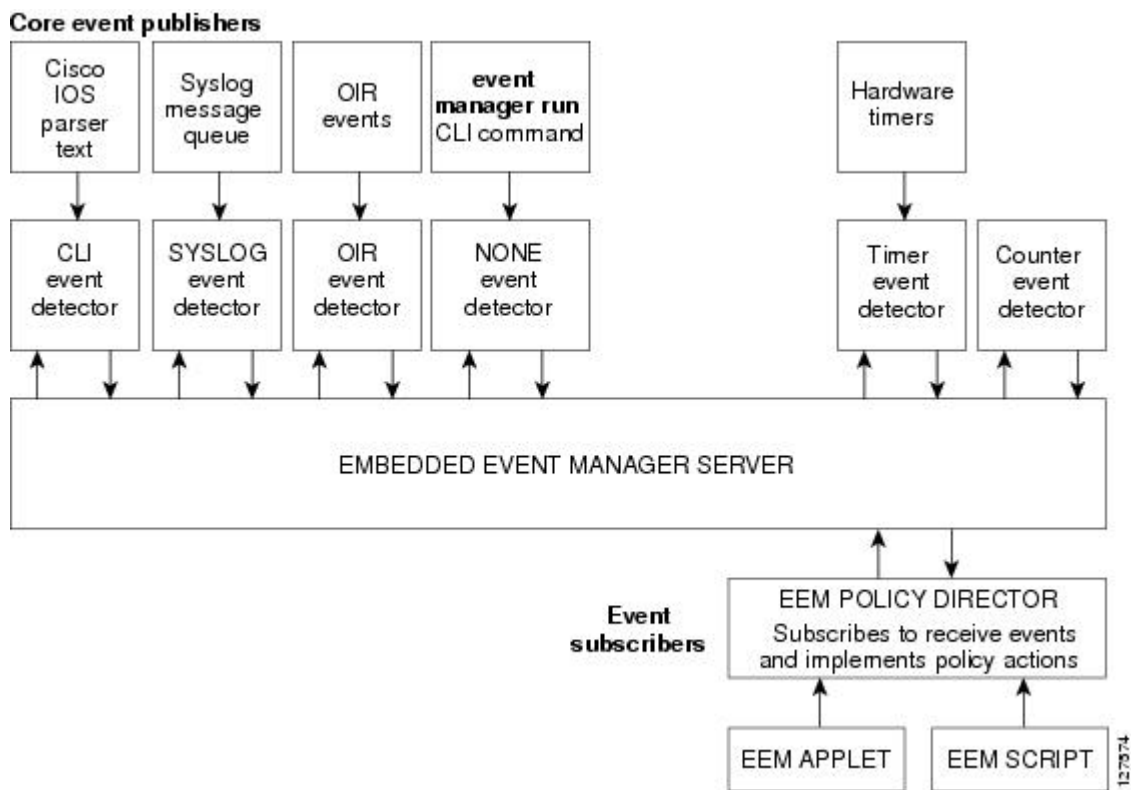
| Category                                                                                                 | Definition                                                                                                                                                                                                                                                                                                                                                                                                                       |
|----------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| EEM event Tcl command extensions (three types: event information, event registration, and event publish) | This category is represented by the <b>event_register_</b> xxx family of event-specific commands. There is a separate event information Tcl command extension in this category as well: <b>event_reqinfo</b> . This is the command used in policies to query the EEM for information about an event. There is also an EEM event publish Tcl command extension <b>event_publish</b> that publishes an application-specific event. |
| EEM action Tcl command extensions                                                                        | These Tcl command extensions (for example, <b>action_syslog</b> ) are used by policies to respond to or recover from an event or fault. In addition to these extensions, developers can use the Tcl language to implement any action desired.                                                                                                                                                                                    |
| EEM utility Tcl command extensions                                                                       | These Tcl command extensions are used to retrieve, save, set, or modify application information, counters, or timers.                                                                                                                                                                                                                                                                                                            |
| EEM system information Tcl command extensions                                                            | This category is represented by the <b>sys_reqinfo_</b> xxx family of system-specific information commands. These commands are used by a policy to gather system information.                                                                                                                                                                                                                                                    |
| EEM context Tcl command extensions                                                                       | These Tcl command extensions are used to store and retrieve a Tcl context (the visible variables and their values).                                                                                                                                                                                                                                                                                                              |

## General Flow of EEM Event Detection and Recovery

EEM is a flexible, policy-driven framework that supports in-box monitoring of different components of the system with the help of software agents known as event detectors. The figure below shows the relationship between the EEM server, the core event publishers (event detectors), and the event subscribers (policies). Basically, event publishers screen events and publish them when there is a match on an event specification that is provided by the event subscriber. Event detectors notify the EEM server when an event of interest occurs.

When an event or fault is detected, Embedded Event Manager determines from the event publishers--an example would be the OIR events publisher in the figure below--if a registration for the encountered fault or event has occurred. EEM matches the event registration information with the event data itself. A policy registers for the detected event with the Tcl command extension `event_register _xxx`. The event information Tcl command extension `event_reqinfo` is used in the policy to query the Embedded Event Manager for information about the detected event.

Figure 142: Embedded Event Manager Core Event Detectors



## Safe-Tcl

Safe-Tcl is a safety mechanism that allows untrusted Tcl scripts to run in an interpreter that was created in the safe mode. The safe interpreter has a restricted set of commands that prevent accessing some system resources and harming the host and other applications. For example, it does not allow commands to access critical Cisco IOS file system directories.

Cisco-defined scripts run in full Tcl mode, but user-defined scripts run in Safe-Tcl mode. Safe-Tcl allows Cisco to disable or customize individual Tcl commands. For more details about Tcl commands, go to <http://www.tcl.tk/man/>.

The following list of Tcl commands are restricted with a few exceptions. Restrictions are noted against each command or command keyword:

- `cd` --Change directory is not allowed to one of the restricted Cisco directory names.



- **encoding** --The commands **encoding names**, **encoding convertfrom**, and **encoding convertto** are permitted. The **encoding system** command with no arguments is permitted, but the **encoding system** command with the **?encoding?** keyword is not permitted.
- **exec** --Not permitted.
- **fconfigure** --Permitted.
- **file** --The following are permitted:
  - **file dirname**
  - **file exists**
  - **file extension**
  - **file isdirectory**
  - **file join**
  - **file pathtype**
  - **file rootname**
  - **file split**
  - **file stat**
  - **file tail**
- **file** --The following are not permitted:
  - **file atime**
  - **file attributes**
  - **file channels**
  - **file copy**
  - **file delete**
  - **file executable**
  - **file isfile**
  - **file link**
  - **file lstat**
  - **file mkdir**
  - **file mtime**
  - **file nativename**
  - **file normalize**
  - **file owned**
  - **file readable**
  - **file readlink**
  - **file rename**
  - **file rootname**
  - **file separator**
  - **file size**
  - **file system**
  - **file type**
  - **file volumes**
  - **file writable**
- **glob** --The **glob** command is not permitted when searching in one of the restricted Cisco directories. Otherwise, it is permitted.

- **load** --Only files that are in the user policy directory or the user library directory are permitted to be loaded. Static packages (for example, libraries that consist of C code) are not permitted to be loaded with the **load** command.
- **open** --The **open** command is not allowed for a file that is located in one of the restricted Cisco directories.
- **pwd** --The **pwd** command is not permitted.
- **socket** --The **socket** command is permitted.
- **source** --The **source** command is permitted for files that are in the user policy directory or the user library directory.

## Bytecode Support for EEM 2.4

EEM 2.4 introduces bytecode language (BCL) support by accepting files with the standard bytecode script extension `.tbc`. Tcl version 8.3.4 defines a BCL and includes a compiler that translates Tcl scripts into BCL. Valid EEM policy file extensions in EEM 2.4 for user and system policies are `.tcl` (Tcl Text files) and `.tbc` (Tcl bytecode files).

Storing Tcl scripts in bytecode improves the execution speed of the policy because the code is precompiled, creates a smaller policy size, and obscures the policy code. Obfuscation makes it a little more difficult to modify scripts and hides logic to preserve intellectual property rights.

Support for bytecode is being added to provide another option for release of supported and trusted code. We recommend that you only run well understood, or trusted and supported software on network devices. To generate Tcl bytecode for IOS EEM support, use TclPro versions 1.4 or 1.5.

To translate a Tcl script to bytecode you can use `procomp`, part of Free TclPro Compiler, or Active State Tcl Development Kit. When a Tcl script is compiled using `procomp`, the code is scrambled and a `.tbc` file is generated. The bytecode files are platform-independent and can be generated on any operating system on which TclPro is available, including Windows, Linux, and UNIX. `Procomp` is part of TclPro and available from <http://www.tcl.tk/software/tclpro>.

## Registration Substitution

In addition to regular Tcl substitution, EEM 2.3 permits the substitution of an individual parameter in an EEM event registration statement line with an environment variable.

EEM 2.4 introduces the ability to replace multiple parameters in event registration statement lines with a single environment variable.




---

**Note** Only the first environment variable supports multiple parameter substitution. Individual parameters can still be specified with additional environment variables after the initial variable.

---

To illustrate the substitution, a single environment variable, `$_eem_syslog_statement` is configured as:

```
::cisco::eem::event_register_syslog pattern COUNT
```

Using the registration substitution, the `$_eem_syslog_statement` environment variable is used in the following EEM user policy:

```
$_eem_syslog_statement occurs $_eem_occurs_val
action_syslog "this is test 3"
```

Environment variables must be defined before a policy using them is registered. To define the `$_eem_syslog_statement` environment variable:

```
Device(config)# event manager environment eem_syslog_statement
::cisco::eem::event_register_syslog pattern COUNT
Device(config)# event manager environment eem_occurs_val 2
```

## Cisco File Naming Convention for EEM

All Embedded Event Manager policy names, policy support files (for example, e-mail template files), and library filenames are consistent with the Cisco file naming convention. In this regard, Embedded Event Manager policy filenames adhere to the following specification:

- An optional prefix--Mandatory.--indicating, if present, that this is a system policy that should be registered automatically at boot time if it is not already registered. For example: Mandatory.sl\_text.tcl.
- A filename body part containing a two-character abbreviation (see the table below) for the first event specified; an underscore part; and a descriptive field part that further identifies the policy.
- A filename suffix part defined as .tcl.

Embedded Event Manager e-mail template files consist of a filename prefix of email\_template, followed by an abbreviation that identifies the usage of the e-mail template.

Embedded Event Manager library filenames consist of a filename body part containing the descriptive field that identifies the usage of the library, followed by \_lib, and a filename suffix part defined as .tcl.

**Table 205: Two-Character Abbreviation Specification**

|    |                            |
|----|----------------------------|
| ap | event_register_appl        |
| cl | event_register_cli         |
| ct | event_register_counter     |
| go | event_register_gold        |
| if | event_register_interface   |
| io | event_register_ioswdsysmon |
| la | event_register_ipsla       |
| nf | event_register_nf          |
| no | event_register_none        |
| oi | event_register_oir         |
| pr | event_register_process     |
| rf | event_register_rf          |

|    |                                  |
|----|----------------------------------|
| rs | event_register_resource          |
| rt | event_register_routing           |
| rp | event_register_rpc               |
| sl | event_register_syslog            |
| sn | event_register_snmp              |
| st | event_register_snmp_notification |
| so | event_register_snmp_object       |
| tm | event_register_timer             |
| tr | event_register_track             |
| ts | event_register_timer_subscriber  |
| wd | event_register_wdsysmon          |

# How to Write Embedded Event Manager Policies Using Tcl

## Registering and Defining an EEM Tcl Script

Perform this task to configure environment variables and register an EEM policy. EEM schedules and runs policies on the basis of an event specification that is contained within the policy itself. When an EEM policy is registered, the software examines the policy and registers it to be run when the specified event occurs.

### Before you begin

You must have a policy available that is written in the Tcl scripting language. Sample policies are provided--see the details in the [Sample EEM Policies, on page 2093](#) to see which policies are available for the Cisco IOS release image that you are using--and these sample policies are stored in the system policy directory.

### SUMMARY STEPS

1. **enable**
2. **show event manager environment** [**all**] *variable-name*
3. **configure terminal**
4. **event manager environment** *variable-name string*
5. Repeat [Registering and Defining an EEM Tcl Script](#) to configure all the environment variables required by the policy to be registered in [Registering and Defining an EEM Tcl Script](#).
6. **event manager policy** *policy-filename* [**type** {**system**|**user**}] [**trap**]
7. **exit**

## DETAILED STEPS

|        | Command or Action                                                                                                                                                                                                    | Purpose                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |
|--------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Step 1 | <b>enable</b><br><b>Example:</b><br><pre>Device&gt; enable</pre>                                                                                                                                                     | Enables privileged EXEC mode. <ul style="list-style-type: none"> <li>• Enter your password if prompted.</li> </ul>                                                                                                                                                                                                                                                                                                                                                                                                              |
| Step 2 | <b>show event manager environment [all] variable-name</b><br><b>Example:</b><br><pre>Device# show event manager environment all</pre>                                                                                | (Optional) Displays the name and value of EEM environment variables. <ul style="list-style-type: none"> <li>• The optional <b>all</b> keyword displays all the EEM environment variables.</li> <li>• The optional <i>variable-name</i> argument displays information about the specified environment variable.</li> </ul>                                                                                                                                                                                                       |
| Step 3 | <b>configure terminal</b><br><b>Example:</b><br><pre>Device# configure terminal</pre>                                                                                                                                | Enters global configuration mode.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |
| Step 4 | <b>event manager environment variable-name string</b><br><b>Example:</b><br><pre>Device(config)# event manager environment _cron_entry 0-59/2 0-23/1 * * 0-6</pre>                                                   | Configures the value of the specified EEM environment variable. <ul style="list-style-type: none"> <li>• In this example, the software assigns a CRON timer environment variable to be set to the second minute of every hour of every day.</li> </ul>                                                                                                                                                                                                                                                                          |
| Step 5 | Repeat <a href="#">Registering and Defining an EEM Tcl Script</a> to configure all the environment variables required by the policy to be registered in <a href="#">Registering and Defining an EEM Tcl Script</a> . | --                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
| Step 6 | <b>event manager policy policy-filename [type {system user}] [trap]</b><br><b>Example:</b><br><pre>Device(config)# event manager policy tm_cli_cmd.tcl type system</pre>                                             | Registers the EEM policy to be run when the specified event defined within the policy occurs. <ul style="list-style-type: none"> <li>• Use the <b>system</b> keyword to register a Cisco-defined system policy.</li> <li>• Use the <b>user</b> keyword to register a user-defined system policy.</li> <li>• Use the <b>trap</b> keyword to generate an SNMP trap when the policy is triggered.</li> <li>• In this example, the sample EEM policy named <code>tm_cli_cmd.tcl</code> is registered as a system policy.</li> </ul> |
| Step 7 | <b>exit</b><br><b>Example:</b><br><pre>Device(config)# exit</pre>                                                                                                                                                    | Exits global configuration mode and returns to privileged EXEC mode.                                                                                                                                                                                                                                                                                                                                                                                                                                                            |

## Examples

In the following example, the **show event manager environment** privileged EXEC command is used to display the name and value of all EEM environment variables.

```
Device# show event manager environment all
No. Name Value
1 _cron_entry 0-59/2 0-23/1 * * 0-6
2 _show_cmd show ver
3 _syslog_pattern .*UPDOWN.*Ethernet1/0.*
4 _config_cmd1 interface Ethernet1/0
5 _config_cmd2 no shut
```

## Displaying EEM Registered Policies

Perform this optional task to display EEM registered policies.

### SUMMARY STEPS

1. **enable**
2. **show event manager policy registered** [**event-type** *event-name*] [**time-ordered**|**name-ordered**] [**detailed** *policy-filename*]

### DETAILED STEPS

#### Step 1 enable

Enables privileged EXEC mode. Enter your password if prompted.

#### Example:

```
Device> enable
```

#### Step 2 show event manager policy registered [event-type *event-name*] [time-ordered|name-ordered] [detailed *policy-filename*]

Use this command with the **time-ordered** keyword to display information about currently registered policies sorted by time, for example:

#### Example:

```
Device# show event manager policy registered time-ordered
No. Type Event Type Trap Time Registered Name
1 system timer cron Off Wed May11 01:43:18 2005 tm_cli_cmd.tcl
 name {crontimer2} cron entry {0-59/1 0-23/1 * * 0-7}
 nice 0 priority normal maxrun 240
2 system syslog Off Wed May11 01:43:28 2005 sl_intf_down.tcl
 occurs 1 pattern {.*UPDOWN.*Ethernet1/0.*}
 nice 0 priority normal maxrun 90
3 system proc abort Off Wed May11 01:43:38 2005 pr_cdp_abort.tcl
 instance 1 path {cdp2.iosproc}
 nice 0 priority normal maxrun 20
```

Use this command with the **name-ordered** keyword to display information about currently registered policies sorted by name, for example:

**Example:**

```
Device# show event manager policy registered name-ordered
No. Type Event Type Trap Time Registered Name
1 system proc abort Off Wed May11 01:43:38 2005 pr_cdp_abort.tcl
 instance 1 path {cdp2.iosproc}
 nice 0 priority normal maxrun 20
2 system syslog Off Wed May11 01:43:28 2005 sl_intf_down.tcl
 occurs 1 pattern {.*UPDOWN.*Ethernet1/0.*}
 nice 0 priority normal maxrun 90
3 system timer cron Off Wed May11 01:43:18 2005 tm_cli_cmd.tcl
 name {crontimer2} cron entry {0-59/1 0-23/1 * * 0-7}
 nice 0 priority normal maxrun 240
```

Use this command with the **event-type** keyword to display information about currently registered policies for the event type specified in the *event-name* argument, for example:

**Example:**

```
Device# show event manager policy registered event-type syslog
No. Type Event Type Time Registered Name
1 system syslog Wed May11 01:43:28 2005 sl_intf_down.tcl
 occurs 1 pattern {.*UPDOWN.*Ethernet1/0.*}
 nice 0 priority normal maxrun 90
```

## Unregistering EEM Policies

Perform this task to remove an EEM policy from the running configuration file. Execution of the policy is canceled.

### SUMMARY STEPS

1. **enable**
2. **show event manager policy registered** [**event-type** *event-name*][**system**| **user**] [**time-ordered**| **name-ordered**] [**detailed** *policy-filename*]
3. **configure terminal**
4. **no event manager policy** *policy-filename*
5. **exit**
6. Repeat [Unregistering EEM Policies](#) to ensure that the policy has been removed.

### DETAILED STEPS

|               | Command or Action                                                                                                                                                                                           | Purpose                                                                                                            |
|---------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------|
| <b>Step 1</b> | <b>enable</b><br><b>Example:</b><br>Device> enable                                                                                                                                                          | Enables privileged EXEC mode. <ul style="list-style-type: none"> <li>• Enter your password if prompted.</li> </ul> |
| <b>Step 2</b> | <b>show event manager policy registered</b> [ <b>event-type</b> <i>event-name</i> ][ <b>system</b>   <b>user</b> ] [ <b>time-ordered</b>   <b>name-ordered</b> ] [ <b>detailed</b> <i>policy-filename</i> ] | (Optional) Displays the EEM policies that are currently registered.                                                |

|               | Command or Action                                                                                                                                                                   | Purpose                                                                                                                                                                                                                                                              |
|---------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|               | <p><b>Example:</b></p> <pre>Device# show event manager policy registered</pre>                                                                                                      | <ul style="list-style-type: none"> <li>The optional <b>system</b> or <b>user</b> keyword displays the registered system or user policies.</li> <li>If no keywords are specified, EEM registered policies for all event types are displayed in time order.</li> </ul> |
| <b>Step 3</b> | <p><b>configure terminal</b></p> <p><b>Example:</b></p> <pre>Device# configure terminal</pre>                                                                                       | Enters global configuration mode.                                                                                                                                                                                                                                    |
| <b>Step 4</b> | <p><b>no event manager policy <i>policy-filename</i></b></p> <p><b>Example:</b></p> <pre>Device(config)# no event manager policy pr_cdp_terminate.tcl</pre>                         | <p>Removes the EEM policy from the configuration, causing the policy to be unregistered.</p> <ul style="list-style-type: none"> <li>In this example, the <b>no</b> form of the command is used to unregister a specified policy.</li> </ul>                          |
| <b>Step 5</b> | <p><b>exit</b></p> <p><b>Example:</b></p> <pre>Device(config)# exit</pre>                                                                                                           | Exits global configuration mode and returns to privileged EXEC mode.                                                                                                                                                                                                 |
| <b>Step 6</b> | <p>Repeat <a href="#">Unregistering EEM Policies</a> to ensure that the policy has been removed.</p> <p><b>Example:</b></p> <pre>Device# show event manager policy registered</pre> | --                                                                                                                                                                                                                                                                   |

## Examples

In the following example, the **show event manager policy registered** privileged EXEC command is used to display the three EEM policies that are currently registered:

```
Device# show event manager policy registered
No. Type Event Type Trap Time Registered Name
1 system timer cron Off Tue Oct11 01:43:18 2005 tm_cli_cmd.tcl
 name {crontimer2} cron entry {0-59/1 0-23/1 * * 0-7}
 nice 0 priority normal maxrun 240.000
2 system syslog Off Tue Oct11 01:43:28 2005 sl_intf_down.tcl
 occurs 1 pattern {.*UPDOWN.*Ethernet1/0.*}
 nice 0 priority normal maxrun 90.000
3 system proc abort Off Tue Oct11 01:43:38 2005 pr_cdp_terminate.tcl
 instance 1 path {cdp2.iosproc}
 nice 0 priority normal maxrun 20.000
```

After the current policies are displayed, it is decided to delete the `pr_cdp_terminate.tcl` policy using the **no** form of the **event manager policy** command:

```
Device# configure terminal
```



```
Device(config)# no event manager policy pr_cdp_terminate.tcl
Device(config)# exit
```

The **show event manager policy registered** privileged EXEC command is entered again to display the EEM policies that are currently registered. The policy `pr_cdp_terminate.tcl` is no longer registered.

```
Device# show event manager policy registered
No. Type Event Type Trap Time Registered Name
1 system timer cron Off Tue Oct11 01:45:17 2005 tm_cli_cmd.tcl
 name {crontimer2} cron entry {0-59/1 0-23/1 * * 0-7}
 nice 0 priority normal maxrun 240.000
2 system syslog Off Tue Oct11 01:45:27 2005 sl_intf_down.tcl
 occurs 1 pattern {.*UPDOWN.*Ethernet1/0.*}
 nice 0 priority normal maxrun 90.000
```

## Suspending EEM Policy Execution

Perform this task to immediately suspend the execution of all EEM policies. Suspending policies, instead of unregistering them, might be necessary for reasons of temporary performance or security.

### SUMMARY STEPS

1. **enable**
2. **show event manager policy registered** [**event-type** *event-name*][**system**| **user**] [**time-ordered**| **name-ordered**] [**detailed** *policy-filename*]
3. **configure terminal**
4. **event manager scheduler suspend**
5. **exit**

### DETAILED STEPS

|               | Command or Action                                                                                                                                                                                                                                                              | Purpose                                                                                                                                                                                                                                                                                                                                      |
|---------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Step 1</b> | <b>enable</b><br><b>Example:</b><br>Device> enable                                                                                                                                                                                                                             | Enables privileged EXEC mode. <ul style="list-style-type: none"> <li>• Enter your password if prompted.</li> </ul>                                                                                                                                                                                                                           |
| <b>Step 2</b> | <b>show event manager policy registered</b> [ <b>event-type</b> <i>event-name</i> ][ <b>system</b>   <b>user</b> ] [ <b>time-ordered</b>   <b>name-ordered</b> ] [ <b>detailed</b> <i>policy-filename</i> ]<br><b>Example:</b><br>Device# show event manager policy registered | (Optional) Displays the EEM policies that are currently registered. <ul style="list-style-type: none"> <li>• The optional <b>system</b> or <b>user</b> keyword displays the registered system or user policies.</li> <li>• If no keywords are specified, EEM registered policies for all event types are displayed in time order.</li> </ul> |
| <b>Step 3</b> | <b>configure terminal</b><br><b>Example:</b><br>Device# configure terminal                                                                                                                                                                                                     | Enters global configuration mode.                                                                                                                                                                                                                                                                                                            |

|               | Command or Action                                                                                            | Purpose                                                              |
|---------------|--------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------|
| <b>Step 4</b> | <b>event manager scheduler suspend</b><br><b>Example:</b><br>Device(config)# event manager scheduler suspend | Immediately suspends the execution of all EEM policies.              |
| <b>Step 5</b> | <b>exit</b><br><b>Example:</b><br>Device(config)# exit                                                       | Exits global configuration mode and returns to privileged EXEC mode. |

### Examples

In the following example, the **show event manager policy registered** privileged EXEC command is used to display all the EEM registered policies:

```
Device# show event manager policy registered
No. Type Event Type Trap Time Registered Name
1 system timer cron Off Sat Oct11 01:43:18 2003 tm_cli_cmd.tcl
 name {crontimer2} cron entry {0-59/1 0-23/1 * * 0-7}
 nice 0 priority normal maxrun 240.000
2 system syslog Off Sat Oct11 01:43:28 2003 sl_intf_down.tcl
 occurs 1 pattern {.*UPDOWN.*Ethernet1/0.*}
 nice 0 priority normal maxrun 90.000
3 system proc abort Off Sat Oct11 01:43:38 2003 pr_cdp_abort.tcl
 instance 1 path {cdp2.iosproc}
 nice 0 priority normal maxrun 20.000
```

The **event manager scheduler suspend** command is entered to immediately suspend the execution of all EEM policies:

```
Device# configure terminal
Device(config)# event manager scheduler suspend
*Nov 2 15:34:39.000: %HA_EM-6-FMS_POLICY_EXEC: fh_io_msg: Policy execution has been
suspended
```

## Managing EEM Policies

Perform this task to specify a directory to use for storing user library files or user-defined EEM policies.



**Note** This task applies only to EEM policies that are written using Tcl scripts.

### SUMMARY STEPS

1. **enable**
2. **show event manager directory user [library| policy]**
3. **configure terminal**
4. **event manager directory user {library path| policy path}**

## 5. exit

## DETAILED STEPS

|        | Command or Action                                                                                                                                                                                                                                  | Purpose                                                                                                                                                                                                                                                                                                                                         |
|--------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Step 1 | <b>enable</b><br><b>Example:</b><br>Device> enable                                                                                                                                                                                                 | Enables privileged EXEC mode. <ul style="list-style-type: none"> <li>• Enter your password if prompted.</li> </ul>                                                                                                                                                                                                                              |
| Step 2 | <b>show event manager directory user [library  policy]</b><br><b>Example:</b><br>Device# show event manager directory user library                                                                                                                 | (Optional) Displays the directory to use for storing EEM user library or policy files. <ul style="list-style-type: none"> <li>• The optional <b>library</b> keyword displays the directory to use for user library files.</li> <li>• The optional <b>policy</b> keyword displays the directory to use for user-defined EEM policies.</li> </ul> |
| Step 3 | <b>configure terminal</b><br><b>Example:</b><br>Device# configure terminal                                                                                                                                                                         | Enters global configuration mode.                                                                                                                                                                                                                                                                                                               |
| Step 4 | <b>event manager directory user {library path  policy path}</b><br><b>Example:</b><br>Device(config)# event manager directory user library disk0:/user_library<br><br>Device(config)# event manager directory user library bootflash:/user_library | Specifies a directory to use for storing user library files or user-defined EEM policies. <ul style="list-style-type: none"> <li>• Use the <i>path</i> argument to specify the absolute pathname to the user directory.</li> </ul>                                                                                                              |
| Step 5 | <b>exit</b><br><b>Example:</b><br>Device(config)# exit                                                                                                                                                                                             | Exits global configuration mode and returns to privileged EXEC mode.                                                                                                                                                                                                                                                                            |

## Examples

In the following example, the **show event manager directory user** privileged EXEC command is used to display the directory, if it exists, to use for storing EEM user library files:

```
Device# show event manager directory user library
disk0:/user_library
```

```
Device# show event manager directory user library
bootflash:/user_library
```

# Modifying History Table Size and Displaying EEM History Data

Perform this optional task to change the size of the history tables and to display EEM history data.

## SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **event manager history size {events | traps} [size]**
4. **exit**
5. **show event manager history events [detailed] [maximum number]**
6. **show event manager history traps [server | policy]**

## DETAILED STEPS

### Step 1 enable

Enables privileged EXEC mode. Enter your password if prompted.

#### Example:

```
Device> enable
```

### Step 2 configure terminal

Enters global configuration mode.

#### Example:

```
Device# configure terminal
```

### Step 3 event manager history size {events | traps} [size]

Use this command to change the size of the EEM event history table or the size of the EEM SNMP trap history table. In the following example, the size of the EEM event history table is changed to 30 entries:

#### Example:

```
Device(config)# event manager history size events 30
```

### Step 4 exit

Exits global configuration mode and returns to privileged EXEC mode.

#### Example:

```
Device(config)# exit
```

### Step 5 show event manager history events [detailed] [maximum number]

Use this command to display information about each EEM event that has been triggered.

#### Example:

```
Device# show event manager history events
No. Time of Event Event Type Name
```

```

1 Fri Sep 9 13:48:40 2005 syslog applet: one
2 Fri Sep 9 13:48:40 2005 syslog applet: two
3 Fri Sep 9 13:48:40 2005 syslog applet: three
4 Fri Sep 9 13:50:00 2005 timer cron script: tm_cli_cmd.tcl
5 Fri Sep 9 13:51:00 2005 timer cron script: tm_cli_cmd.tcl

```

**Step 6** `show event manager history traps [server | policy]`

Use this command to display the EEM SNMP traps that have been sent either from the EEM server or from an EEM policy.

**Example:**

```

Device# show event manager history traps
No. Time Trap Type Name
1 Fri Sep 9 13:48:40 2005 server applet: four
2 Fri Sep 9 13:57:03 2005 policy script: no_snmp_test.tcl

```

## Displaying Software Modularity Process Reliability Metrics Using EEM

Perform this optional task to display reliability metrics for Cisco IOS Software Modularity processes. The `show event manager metric processes` command is supported only in Software Modularity images.

**SUMMARY STEPS**

1. `enable`
2. `show event manager metric process {all|process-name}`

**DETAILED STEPS****Step 1** `enable`

Enables privileged EXEC mode. Enter your password if prompted.

**Example:**

```
Device> enable
```

**Step 2** `show event manager metric process {all|process-name}`

Use this command to display the reliability metric data for processes. The system keeps a record of when processes start and end, and this data is used as the basis for reliability analysis. In this partial example, the first and last entries showing the metric data for the processes on all the cards inserted in the system are displayed.

**Example:**

```

Device# show event manager metric process all
=====
process name: devc-pty, instance: 1
sub_system id: 0, version: 00.00.0000

last event type: process start
recent start time: Fri Oct10 20:34:40 2005
recent normal end time: n/a

```

```

recent abnormal end time: n/a
number of times started: 1
number of times ended normally: 0
number of times ended abnormally: 0
most recent 10 process start times:

Fri Oct10 20:34:40 2005

most recent 10 process end times and types:
cumulative process available time: 6 hours 30 minutes 7 seconds 378 milliseconds
cumulative process unavailable time: 0 hours 0 minutes 0 seconds 0 milliseconds
process availability: 0.100000000
number of abnormal ends within the past 60 minutes (since reload): 0
number of abnormal ends within the past 24 hours (since reload): 0
number of abnormal ends within the past 30 days (since reload): 0
.
.
.
=====
process name: cdp2.iosproc, instance: 1
sub_system id: 0, version: 00.00.0000

last event type: process start
recent start time: Fri Oct10 20:35:02 2005
recent normal end time: n/a
recent abnormal end time: n/a
number of times started: 1
number of times ended normally: 0
number of times ended abnormally: 0
most recent 10 process start times:

Fri Oct10 20:35:02 2005

most recent 10 process end times and types:

cumulative process available time: 6 hours 29 minutes 45 seconds 506 milliseconds
cumulative process unavailable time: 0 hours 0 minutes 0 seconds 0 milliseconds
process availability: 0.100000000
number of abnormal ends within the past 60 minutes (since reload): 0
number of abnormal ends within the past 24 hours (since reload): 0
number of abnormal ends within the past 30 days (since reload): 0

```

## Troubleshooting Tips

Use the **debug event manager** command in privileged EXEC mode to troubleshoot EEM command operations. Use any debugging command with caution because the volume of output generated can slow or stop the device operations. We recommend that this command be used only under the supervision of a Cisco engineer.

## Modifying the Sample EEM Policies

Perform this task to modify one of the sample policies. Cisco software contains some sample policies in the images that contain the Embedded Event Manager. Developers of EEM policies may modify these policies by customizing the event for which the policy is to be run and the options associated with logging and responding to the event. In addition, developers may select the actions to be implemented when the policy runs.

## Sample EEM Policies

Cisco includes a set of sample policies shown in the table below. You can copy the sample policies to a user directory and then modify the policies, or you can write your own policies. Tcl is currently the only Cisco-supported scripting language for policy creation. Tcl policies can be modified using a text editor such as Emacs. Policies must execute within a defined number of seconds of elapsed time, and the time variable can be configured within a policy. The default is currently 20 seconds.

The table below describes the sample EEM policies.

**Table 206: Sample EEM Policy Descriptions**

| Name of Policy         | Description                                                                                                                                                                                                                                                                                                                                         |
|------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| pr_cdp_abort.tcl       | Introduced with Cisco Software Modularity images. This policy monitors for cdp2.iosproc process termination events. It will log a message to SYSLOG and send an e-mail with the details of the termination.                                                                                                                                         |
| pr_crash_reporter.tcl  | Introduced with Cisco Software Modularity images. This policy monitors for all process termination events. When an event occurs, the policy will send crash information, including the crashdump file, to the specified URL where a CGI script processes the data.                                                                                  |
| pr_iprouting_abort.tcl | Introduced with Cisco Software Modularity images. This policy monitors for iprouting.iosproc process termination events. It will log a message to SYSLOG and send an e-mail with the details of the termination.                                                                                                                                    |
| sl_intf_down.tcl       | This policy runs when a configurable syslog message is logged. It will execute a configurable CLI command and e-mail the results.                                                                                                                                                                                                                   |
| tm_cli_cmd.tcl         | This policy runs using a configurable CRON entry. It will execute a configurable CLI command and e-mail the results.                                                                                                                                                                                                                                |
| tm_crash_history.tcl   | Introduced with Cisco Software Modularity images. This policy runs at midnight every day and e-mails a process crash history report to a specified e-mail address.                                                                                                                                                                                  |
| tm_crash_reporter.tcl  | This policy runs 5 seconds after it is registered. If the policy is saved in the configuration, it will also run each time that the device is reloaded. The policy will prompt for the reload reason. If the reload was due to a crash, the policy will search for the latest crashinfo file and send this information to a specified URL location. |
| tm_fsyz_usage.tcl      | Introduced with Cisco Software Modularity images. This policy runs using a configurable CRON entry and monitors disk space usage. A syslog message will be displayed if disk space usage crosses configurable thresholds.                                                                                                                           |
| wd_mem_reporter.tcl    | Introduced with Cisco Software Modularity images. This policy reports on low system memory conditions when the amount of memory available falls below 20 percent of the initial available system memory. A syslog message will be displayed and, optionally, an e-mail will be sent.                                                                |

For more details about the sample policies available and how to run them, see the [EEM Event Detector Demo Examples, on page 2111](#).

## SUMMARY STEPS

1. **enable**
2. **show event manager policy available detailed** *policy-filename*
3. Cut and paste the contents of the sample policy displayed on the screen to a text editor.
4. Edit the policy and save it with a new filename.
5. Copy the new file back to the device flash memory.
6. **configure terminal**
7. **event manager directory user** {*library path*|*policy path*}
8. **event manager policy** *policy-filename* [**type** {*system*|*user*}] [**trap**]

## DETAILED STEPS

### Step 1 enable

Enables privileged EXEC mode. Enter your password if prompted.

#### Example:

```
Device> enable
```

### Step 2 show event manager policy available detailed *policy-filename*

Displays the actual specified sample policy including details about the environment variables used by the policy and instructions for running the policy. The **detailed** keyword was introduced for the **show event manager policy available** and the **show event manager policy registered** commands. Depending on your release, you may need to copy one of the two Tcl scripts from the configuration examples section in this document (see the [Programming Policies with Tcl Sample Scripts Example, on page 2119](#)). In the following example, details about the sample policy `tm_cli_cmd.tcl` are displayed on the screen.

#### Example:

```
Device# show event manager policy available detailed tm_cli_cmd.tcl
```

### Step 3 Cut and paste the contents of the sample policy displayed on the screen to a text editor.

Use the edit and copy functions to move the contents from the device to a text editor on another device.

### Step 4 Edit the policy and save it with a new filename.

Use the text editor to modify the policy as a Tcl script. For file naming conventions, see the [Cisco File Naming Convention for EEM, on page 2081](#).

### Step 5 Copy the new file back to the device flash memory.

Copy the file to the flash file system on the device--typically `disk0:`. For more details about copying files, see the "Using the Cisco IOS File System" chapter in the *Configuration Fundamentals Configuration Guide*.

Copy the file to the flash file system on the device--typically `bootflash:`. For more details about copying files, see the "Using the Cisco IOS File System" chapter in the *Configuration Fundamentals Configuration Guide*.

### Step 6 configure terminal

Enters global configuration mode.

#### Example:



```
Device# configure terminal
```

**Step 7** `event manager directory user {library path} policy path}`

Specifies a directory to use for storing user library files or user-defined EEM policies. In the following example, the `user_library` directory on `disk0` is specified as the directory for storing user library files.

Specifies a directory to use for storing user library files or user-defined EEM policies. In the following example, the `user_library` directory on `bootflash` is specified as the directory for storing user library files.

**Example:**

```
Device(config)# event manager directory user library disk0:/user_library
```

```
Device(config)# event manager directory user library bootflash:/user_library
```

**Step 8** `event manager policy policy-filename [type {system| user}] [trap]`

Registers the EEM policy to be run when the specified event defined within the policy occurs. In the following example, the new EEM policy named `test.tcl` is registered as a user-defined policy.

**Example:**

```
Device(config)# event manager policy test.tcl type user
```

---

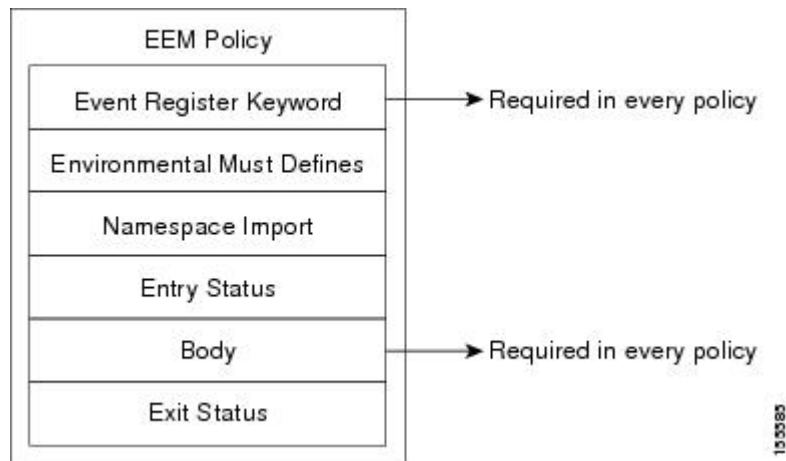
## Programming EEM Policies with Tcl

Perform this task to help you program a policy using Tcl command extensions. We recommend that you copy an existing policy and modify it. There are two required parts that must exist in an EEM Tcl policy: the `event_register` Tcl command extension and the body. All other sections shown in the [Tcl Policy Structure and Requirements, on page 2095](#) concept are optional.

### Tcl Policy Structure and Requirements

All EEM policies share the same structure, shown in the figure below. There are two parts of an EEM policy that are required: the `event_register` Tcl command extension and the body. The remaining parts of the policy are optional: environment must defines, namespace import, entry status, and exit status.

Figure 143: Tcl Policy Structure and Requirements



The start of every policy must describe and register the event to detect using an **event\_register** Tcl command extension. This part of the policy schedules the running of the policy. The following example Tcl code shows how to register the **event\_register\_timer** Tcl command extension:

```
::cisco::eem::event_register_timer cron name crontimer2 cron_entry $_cron_entry maxrun 240
```

The environment must defines section is optional and includes the definition of environment variables. The following example Tcl code shows how to check for, and define, some environment variables.

```
Check if all the env variables that we need exist.
If any of them does not exist, print out an error msg and quit.
if {[info exists _email_server]} {
 set result \
 "Policy cannot be run: variable _email_server has not been set"
 error $result $errorInfo
}
if {[info exists _email_from]} {
 set result \
 "Policy cannot be run: variable _email_from has not been set"
 error $result $errorInfo
}
if {[info exists _email_to]} {
 set result \
 "Policy cannot be run: variable _email_to has not been set"
 error $result $errorInfo
}
```

The namespace import section is optional and defines code libraries. The following example Tcl code shows how to configure a namespace import section.

```
namespace import ::cisco::eem::*
namespace import ::cisco::lib::*
```

The body of the policy is a required structure and might contain the following:

- The **event\_reqinfo** event information Tcl command extension that is used to query the EEM for information about the detected event.
- The action Tcl command extensions, such as **action\_syslog**, that are used to specify EEM specific actions.

- The system information Tcl command extensions, such as **sys\_reqinfo\_routername**, that are used to obtain general system information.
- Use of the SMTP library (to send e-mail notifications) or the CLI library (to run CLI commands) from a policy.
- The **context\_save** and **context\_retrieve** Tcl command extensions that are used to save Tcl variables for use by other policies.

The following example Tcl code shows the code to query an event and log a message as part of the body section.

```
Query the event info and log a message.
array set arr_einfo [event_reqinfo]

if {$_cerrno != 0} {
 set result [format "component=%s; subsys err=%s; posix err=%s;\n%s" \
 $_cerr_sub_num $_cerr_sub_err $_cerr_posix_err $_cerr_str]
 error $result
}

global timer_type timer_time_sec
set timer_type $arr_einfo(timer_type)
set timer_time_sec $arr_einfo(timer_time_sec)

Log a message.
set msg [format "timer event: timer type %s, time expired %s" \
 $timer_type [clock format $timer_time_sec]]

action_syslog priority info msg $msg
if {$_cerrno != 0} {
 set result [format "component=%s; subsys err=%s; posix err=%s;\n%s" \
 $_cerr_sub_num $_cerr_sub_err $_cerr_posix_err $_cerr_str]
 error $result
}
```

## EEM Entry Status

The entry status part of an EEM policy is used to determine if a prior policy has been run for the same event, and to determine the exit status of the prior policy. If the `_entry_status` variable is defined, a prior policy has already run for this event. The value of the `_entry_status` variable determines the return code of the prior policy.

Entry status designations may use one of three possible values: 0 (previous policy was successful), Not=0 (previous policy failed), and Undefined (no previous policy was executed).

## EEM Exit Status

When a policy finishes running its code, an exit value is set. The exit value is used by the Embedded Event Manager to determine whether or not to apply the default action for this event, if any. A value of zero means do not perform the default action. A value of nonzero means perform the default action. The exit status will be passed to subsequent policies that are run for the same event.

## EEM Policies and Cisco Error Number

Some EEM Tcl command extensions set a Cisco Error Number Tcl global variable `_cerrno`. Whenever `_cerrno` is set, four other Tcl global variables are derived from `_cerrno` and are set along with it (`_cerr_sub_num`, `_cerr_sub_err`, `_cerr_posix_err`, and `_cerr_str`).

For example, the **action\_syslog** command in the example below sets these global variables as a side effect of the command execution:

```
action_syslog priority warning msg "A sample message generated by action_syslog"
if {$_cerrno != 0} {
 set result [format "component=%s; subsys err=%s; posix err=%s;\n%s" \
 $_cerr_sub_num $_cerr_sub_err $_cerr_posix_err $_cerr_str]
 error $result
}
```

### `_cerrno`: 32-Bit Error Return Values

The `_cerrno` set by a command can be represented as a 32-bit integer of the following form:

```
XYSSSSSSSSSSSSSEEEEEEEEEPPPPPPPP
```

For example, the following error return value might be returned from an EEM Tcl command extension:

```
862439AE
```

This number is interpreted as the following 32-bit value:

```
10000110001001000011100110101110
```

This 32-bit integer is divided up into the five variables shown in the table below.

**Table 207: `_cerrno`: 32-Bit Error Return Value Variables**

| Variable       | Description                                                                                                                                                                                                                                                                                          |
|----------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| XY             | The error class (indicates the severity of the error). This variable corresponds to the first two bits in the 32-bit error return value; 10 in the case above, which indicates CERR_CLASS_WARNING:<br><br>See the table below for the four possible error class encodings specific to this variable. |
| SSSSSSSSSSSSSS | The subsystem number that generated the most recent error (13 bits = 8192 values). This is the next 13 bits of the 32-bit sequence, and its integer value is contained in <code>\$_cerr_sub_num</code> .                                                                                             |
| Variable       | Description                                                                                                                                                                                                                                                                                          |
| EEEEEEEE       | The subsystem specific error number (8 bits = 256 values). This segment is the next 8 bits of the 32-bit sequence, and the string corresponding to this error number is contained in <code>\$_cerr_sub_err</code> .                                                                                  |
| PPPPPPPP       | The pass-through POSIX error code (9 bits = 512 values). This represents the last of the 32-bit sequence, and the string corresponding to this error code is contained in <code>\$_cerr_posix_err</code> .                                                                                           |

### Error Class Encodings for XY

The first variable, XY, references the possible error class encodings shown in the table below.

**Table 208: Error Class Encodings**

|    |                    |
|----|--------------------|
| 00 | CERR_CLASS_SUCCESS |
| 01 | CERR_CLASS_INFO    |
| 10 | CERR_CLASS_WARNING |
| 11 | CERR_CLASS_FATAL   |

An error return value of zero means SUCCESS.

## SUMMARY STEPS

1. **enable**
2. **show event manager policy available detailed** *policy-filename*
3. Cut and paste the contents of the sample policy displayed on the screen to a text editor.
4. Define the required **event\_register** Tcl command extension.
5. Add the appropriate namespace under the `::cisco` hierarchy.
6. Program the `must defines` section to check for each environment variable that is used in this policy.
7. Program the body of the script.
8. Check the entry status to determine if a policy has previously run for this event.
9. Check the exit status to determine whether or not to apply the default action for this event, if a default action exists.
10. Set Cisco Error Number (`_cerno`) Tcl global variables.
11. Save the Tcl script with a new filename, and copy the Tcl script to the device.
12. **configure terminal**
13. **event manager directory user** *{library path| policy path}*
14. **event manager policy** *policy-filename* [**type** *{system| user}*] [**trap**]
15. Cause the policy to execute, and observe the policy.
16. Use debugging techniques if the policy does not execute correctly.

## DETAILED STEPS

### Step 1

**enable**

Enables privileged EXEC mode. Enter your password if prompted.

**Example:**

```
Device> enable
```

### Step 2

**show event manager policy available detailed** *policy-filename*

Displays the actual specified sample policy including details about the environment variables used by the policy and instructions for running the policy. The **detailed** keyword was introduced for the **show event manager policy available** and the **show event manager policy registered** commands. Depending on your release, you must copy one of the two Tcl scripts from the configuration examples section in this document (see the [Programming Policies with Tcl Sample](#)

[Scripts Example, on page 2119](#)). In the following example, details about the sample policy `tm_cli_cmd.tcl` are displayed on the screen.

**Example:**

```
Device# show event manager policy available detailed tm_cli_cmd.tcl
```

**Step 3** Cut and paste the contents of the sample policy displayed on the screen to a text editor.

Use the edit and copy functions to move the contents from the device to a text editor on another device. Use the text editor to edit the policy as a Tcl script.

**Step 4** Define the required **event\_register** Tcl command extension.

Choose the appropriate **event\_register** Tcl command extension from the table below for the event that you want to detect, and add it to the policy.

*Table 209: EEM Event Registration Tcl Command Extensions*

| Event Registration Tcl Command Extensions |
|-------------------------------------------|
| event_register_appl                       |
| event_register_cli                        |
| event_register_counter                    |
| event_register_gold                       |
| event_register_interface                  |
| event_register_ioswdsysmon                |
| event_register_ipsla                      |
| event_register_nf                         |
| event_register_none                       |
| event_register_oir                        |
| event_register_process                    |
| event_register_resource                   |
| event_register_rf                         |
| event_register_routing                    |
| event_register_rpc                        |
| event_register_snmp                       |
| event_register_snmp_notification          |
| event_register_snmp_object                |
| event_register_syslog                     |

| Event Registration Tcl Command Extensions |
|-------------------------------------------|
| event_register_timer                      |
| event_register_timer_subscriber           |
| event_register_track                      |
| event_register_wdysysmon                  |

**Step 5** Add the appropriate namespace under the ::cisco hierarchy.

Policy developers can use the new namespace ::cisco in Tcl policies in order to group all the extensions used by Cisco IOS EEM. There are two namespaces under the ::cisco hierarchy, and the table below shows which category of EEM Tcl command extension belongs under each namespace.

**Table 210: Cisco IOS EEM Namespace Groupings**

| Namespace    | Category of Tcl Command Extension |
|--------------|-----------------------------------|
| ::cisco::eem | EEM event registration            |
|              | EEM event information             |
|              | EEM event publish                 |
|              | EEM action                        |
|              | EEM utility                       |
|              | EEM context library               |
|              | EEM system information            |
|              | CLI library                       |
| ::cisco::lib | SMTP library                      |

**Note** Make sure that you import the appropriate namespaces or use the qualified command names when using the above commands.

**Step 6** Program the must defines section to check for each environment variable that is used in this policy.

This is an optional step. Must defines are a section of the policy that tests whether any EEM environment variables that are required by the policy are defined before the recovery actions are taken. The must defines section is not required if the policy does not use any EEM environment variables. EEM environment variables for EEM scripts are Tcl global variables that are defined external to the policy before the policy is run. To define an EEM environment variable, use the Embedded Event Manager configuration command **event manager environment** CLI command. By convention all Cisco EEM environment variables begin with “\_” (an underscore). In order to avoid future conflict, customers are urged not to define new variables that start with “\_”.

**Note** You can display the Embedded Event Manager environment variables set on your system by using the **show event manager environment** privileged EXEC command.

For example, Embedded Event Manager environment variables defined by the sample policies include e-mail variables. The sample policies that send e-mail must have the variables shown in the table below set in order to function properly.

The table below describes the e-mail-specific environment variables used in the sample EEM policies.

**Table 211: E-mail-Specific Environmental Variables Used by the Sample Policies**

| Environment Variable | Description                                                             | Example                                                                                                                                                                                         |
|----------------------|-------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| _email_server        | A Simple Mail Transfer Protocol (SMTP) mail server used to send e-mail. | The e-mail server name can be in any one of the following template formats: <ul style="list-style-type: none"> <li>• username:password@host</li> <li>• username@host</li> <li>• host</li> </ul> |
| _email_to            | The address to which e-mail is sent.                                    | engineering@example.com                                                                                                                                                                         |
| _email_from          | The address from which e-mail is sent.                                  | devtest@example.com                                                                                                                                                                             |
| _email_cc            | The address to which the e-mail must be copied.                         | manager@example.com                                                                                                                                                                             |

The following example of a must define section shows how to program a check for e-mail-specific environment variables.

### Example of Must Defines

#### Example:

```

if {[info exists _email_server]} {
 set result \
 "Policy cannot be run: variable _email_server has not been set"
 error $result $errorInfo
}
if {[info exists _email_from]} {
 set result \
 "Policy cannot be run: variable _email_from has not been set"
 error $result $errorInfo
}
if {[info exists _email_to]} {
 set result \
 "Policy cannot be run: variable _email_to has not been set"
 error $result $errorInfo
}
if {[info exists _email_cc]} {
 set result \
 "Policy cannot be run: variable _email_cc has not been set"
 error $result $errorInfo
}

```

### Step 7

Program the body of the script.

In this section of the script, you can define any of the following:

- The **event\_reqinfo** event information Tcl command extension that is used to query the EEM for information about the detected event.
- The action Tcl command extensions, such as **action\_syslog**, that are used to specify EEM specific actions.



- The system information Tcl command extensions, such as **sys\_reqinfo\_routername**, that are used to obtain general system information.
- The **context\_save** and **context\_retrieve** Tcl command extensions that are used to save Tcl variables for use by other policies.
- Use of the SMTP library (to send e-mail notifications) or the CLI library (to run CLI commands) from a policy.

**Step 8** Check the entry status to determine if a policy has previously run for this event.

If the prior policy is successful, the current policy may or may not require execution. Entry status designations may use one of three possible values: 0 (previous policy was successful), Not=0 (previous policy failed), and Undefined (no previous policy was executed).

**Step 9** Check the exit status to determine whether or not to apply the default action for this event, if a default action exists.

A value of zero means do not perform the default action. A value of nonzero means perform the default action. The exit status will be passed to subsequent policies that are run for the same event.

**Step 10** Set Cisco Error Number (`_cerrno`) Tcl global variables.

Some EEM Tcl command extensions set a Cisco Error Number Tcl global variable `_cerrno`. Whenever `_cerrno` is set, four other Tcl global variables are derived from `_cerrno` and are set along with it (`_cerr_sub_num`, `_cerr_sub_err`, `_cerr_posix_err`, and `_cerr_str`).

For example, the **action\_syslog** command in the example below sets these global variables as a side effect of the command execution:

**Example:**

```
action_syslog priority warning msg "A sample message generated by action_syslog
if {$_cerrno != 0} {
 set result [format "component=%s; subsys err=%s; posix err=%s;\n%s" \
 $_cerr_sub_num $_cerr_sub_err $_cerr_posix_err $_cerr_str]
 error $result
}
```

**Step 11** Save the Tcl script with a new filename, and copy the Tcl script to the device.

Embedded Event Manager policy filenames adhere to the following specification:

- An optional prefix--Mandatory.--indicating, if present, that this is a system policy that should be registered automatically at boot time if it is not already registered. For example: Mandatory.sl\_text.tcl.
- A filename body part containing a two-character abbreviation (see [EEM Policies and Cisco Error Number, on page 2098](#)) for the first event specified; an underscore character part; and a descriptive field part further identifying the policy.
- A filename suffix part defined as .tcl.

For more details, see the [Cisco File Naming Convention for EEM, on page 2081](#).

Copy the file to the flash file system on the device--typically disk0:. For more details about copying files, see the "Using the Cisco IOS File System" chapter in the Cisco IOS Configuration Fundamentals Configuration Guide .

Copy the file to the flash file system on the device--typically bootflash:. For more details about copying files, see the "Using the Cisco IOS File System" chapter in the Cisco IOS Configuration Fundamentals Configuration Guide .

**Step 12** **configure terminal**

Enters global configuration mode.

**Example:**

```
Device# configure terminal
```

**Step 13** **event manager directory user** *{library path| policy path}*

Specifies a directory to use for storing user library files or user-defined EEM policies. In the following example, the `user_library` directory on `disk0` is specified as the directory for storing user library files.

Specifies a directory to use for storing user library files or user-defined EEM policies. In the following example, the `user_library` directory on `bootflash` is specified as the directory for storing user library files.

**Example:**

```
Device(config)# event manager directory user library disk0:/user_library
```

```
Device(config)# event manager directory user library bootflash:/user_library
```

**Step 14** **event manager policy** *policy-filename* [**type** *{system| user}*] [**trap**]

Registers the EEM policy to be run when the specified event defined within the policy occurs. In the following example, the new EEM policy named `cl_mytest.tcl` is registered as a user-defined policy.

**Example:**

```
Device(config)# event manager policy cl_mytest.tcl type user
```

**Step 15** Cause the policy to execute, and observe the policy.

To test that the policy runs, generate the conditions that will cause the policy to execute and observe that the policy runs as expected.

**Step 16** Use debugging techniques if the policy does not execute correctly.

Use the Cisco IOS **debug event manager** CLI command with its various keywords to debug issues. Refer to the [Troubleshooting Tips, on page 2104](#) for details about using Tcl-specific keywords.

## Troubleshooting Tips

- Use the **debug event manager tcl commands** CLI command to debug issues with Tcl extension commands. When enabled, this command displays all data that is passed in and read back from the TTY session that handles the CLI interactions. This data helps ensure users that the commands they are passing to the CLI are valid.
- The CLI library allows users to run CLI commands and obtain the output of commands in Tcl. Use the **debug event manager tcl cli-library** CLI command to debug issues with the CLI library.
- The SMTP library allows users to send e-mail messages to an SMTP e-mail server. Use the **debug event manager tcl smtp\_library** CLI command to debug issues with the SMTP library. When enabled, this command displays all data that is passed in and read back from the SMTP library routines. This data helps ensure users that the commands they are passing to the SMTP library are valid.
- Tcl is a flexible language that allows you to override commands. For example, you can modify the **set** command and create a version of the **set** command that displays a message when a scalar variable is set.

When the **set** command is entered in a policy, a message is displayed anytime a scalar variable is set, and this provides a way to debug scalar variables. To view an example of this debugging technique, see the [Tracing Tcl set Command Operations Example, on page 2130](#).

To view examples of the some of these debugging techniques, see the [Debugging Embedded Event Manager Policies Examples, on page 2128](#).

## Creating an EEM User Tcl Library Index

Perform this task to create an index file that contains a directory of all the procedures contained in a library of Tcl files. This task allows you to test library support in EEM Tcl. In this task, a library directory is created to contain the Tcl library files, the files are copied into the directory, and an index (tclIndex) is created that contains a directory of all the procedures in the library files. If the index is not created, the Tcl procedures will not be found when an EEM policy is run that references a Tcl procedure.

### SUMMARY STEPS

1. On your workstation (UNIX, Linux, PC, or Mac) create a library directory and copy the Tcl library files into the directory.
2. **telsh**
3. **auto\_mkindex** *directory\_name* \*.tcl
4. Copy the Tcl library files and the tclIndex file to the directory used for storing user library files on the target device.
5. Copy a user-defined EEM policy file written in Tcl to the directory used for storing user-defined EEM policies on the target device.
6. **enable**
7. **configure terminal**
8. **event manager directory user library** *path*
9. **event manager directory user policy** *path*
10. **event manager policy** *policy-name* [type {system | user}] [trap ]
11. **event manager run** *policy-name*

### DETAILED STEPS

**Step 1** On your workstation (UNIX, Linux, PC, or Mac) create a library directory and copy the Tcl library files into the directory.

The following example files can be used to create a tclIndex on a workstation running the Tcl shell:

**lib1.tcl**

**Example:**

```
proc test1 {} {
 puts "In procedure test1"
}

proc test2 {} {
 puts "In procedure test2"
}
```

**lib2.tcl****Example:**

```
proc test3 {} {
 puts "In procedure test3"
}
```

**Step 2** **tclsh**

Use this command to enter the Tcl shell.

**Example:**

```
workstation% tclsh
```

**Step 3** **auto\_mkindex** *directory\_name* \*.tcl

Use the **auto\_mkindex** command to create the tclIndex file. The tclIndex file that contains a directory of all the procedures contained in the Tcl library files. We recommend that you run auto\_mkindex inside a directory because there can only be a single tclIndex file in any directory and you may have other Tcl files to be grouped together. Running auto\_mkindex in a directory determines which tcl source file or files are indexed using a specific tclIndex.

**Example:**

```
workstation% auto_mkindex eem_library *.tcl
```

The following example TclIndex is created when the lib1.tcl and lib2.tcl files are in a library file directory and the **auto\_mkindex** command is run.

**tclIndex****Example:**

```
Tcl autoload index file, version 2.0
This file is generated by the "auto_mkindex" command
and sourced to set up indexing information for one or
more commands. Typically each line is a command that
sets an element in the auto_index array, where the
element name is the name of a command and the value is
a script that loads the command.

set auto_index(test1) [list source [file join $dir lib1.tcl]]
set auto_index(test2) [list source [file join $dir lib1.tcl]]
set auto_index(test3) [list source [file join $dir lib2.tcl]]
```

**Step 4** Copy the Tcl library files and the tclIndex file to the directory used for storing user library files on the target device.

**Step 5** Copy a user-defined EEM policy file written in Tcl to the directory used for storing user-defined EEM policies on the target device.

The directory for storing user-defined EEM policies can be the same directory used in Step 4. The following example user-defined EEM policy can be used to test the Tcl library support in EEM.

**libtest.tcl****Example:**

```
::cisco::eem::event_register_none

namespace import ::cisco::eem::*
namespace import ::cisco::lib::*
```

```

global auto_index auto_path

puts [array names auto_index]

if { [catch {test1} result]} {
 puts "calling test1 failed result = $result $auto_path"
}

if { [catch {test2} result]} {
 puts "calling test2 failed result = $result $auto_path"
}
if { [catch {test3} result]} {
 puts "calling test3 failed result = $result $auto_path"
}

```

**Step 6**    **enable**

Enables privileged EXEC mode. Enter your password if prompted.

**Example:**

```
Device> enable
```

**Step 7**    **configure terminal**

Enables global configuration mode.

**Example:**

```
Device# configure terminal
```

**Step 8**    **event manager directory user library *path***

Use this command to specify the EEM user library directory; this is the directory to which the files were copied.

**Example:**

```
Device(config)# event manager directory user library disk2:/eem_library
```

**Step 9**    **event manager directory user policy *path***

Use this command to specify the EEM user policy directory; this is the directory to which the file was copied.

**Example:**

```
Device(config)# event manager directory user policy disk2:/eem_policies
```

**Step 10**    **event manager policy *policy-name* [type {system | user}] [trap ]**

Use this command to register a user-defined EEM policy. In this example, the policy named libtest.tcl is registered.

**Example:**

```
Device(config)# event manager policy libtest.tcl
```

**Step 11**    **event manager run *policy-name***

Use this command to manually run an EEM policy. In this example, the policy named libtest.tcl is run to test the Tcl support in EEM. The example output shows that the test for Tcl support in EEM was successful.

**Example:**

```
Device(config)# event manager run libtest.tcl
The following output is displayed:
01:24:37: %HA_EM-6-LOG: libtest.tcl: In procedure test1
01:24:37: %HA_EM-6-LOG: libtest.tcl: In procedure test2
01:24:37: %HA_EM-6-LOG: libtest.tcl: In procedure test3
```

## Creating an EEM User Tcl Package Index

Perform this task to create a Tcl package index file that contains a directory of all the Tcl packages and version information contained in a library of Tcl package files. Tcl packages are supported, depending on your release, using the Tcl **package** keyword.

Tcl packages are located in either the EEM system library directory or the EEM user library directory. When a **package require** Tcl command is executed, the user library directory is searched first for a pkgIndex.tcl file. If the pkgIndex.tcl file is not found in the user directory, the system library directory is searched. In this task, a Tcl package directory--the pkgIndex.tcl file--is created in the appropriate library directory using the **pkg\_mkIndex** command to contain information about all of the Tcl packages contained in the directory along with version information. If the index is not created, the Tcl packages will not be found when an EEM policy is run that contains a **package require** Tcl command.

Using the Tcl package support in EEM, users can gain access to packages such as XML\_RPC for Tcl. When the Tcl package index is created, a Tcl script can easily make an XML-RPC call to an external entity.




---

**Note** Packages implemented in C programming code are not supported in EEM.

---

### SUMMARY STEPS

1. On your workstation (UNIX, Linux, PC, or Mac) create a library directory and copy the Tcl package files into the directory.
2. **tclsh**
3. **pkg\_mkindex** *directory\_name* \*.tcl
4. Copy the Tcl library files and the pkgIndex file to the directory used for storing user library files on the target device.
5. Copy a user-defined EEM policy file written in Tcl to the directory used for storing user-defined EEM policies on the target device.
6. **enable**
7. **configure terminal**
8. **event manager directory user library** *path*
9. **event manager directory user policy** *path*
10. **event manager policy** *policy-name* [**type** {system | user}] [**trap**]
11. **event manager run** *policy-name*

## DETAILED STEPS

---

**Step 1** On your workstation (UNIX, Linux, PC, or Mac) create a library directory and copy the Tcl package files into the directory.

**Step 2** **tclsh**

Use this command to enter the Tcl shell.

**Example:**

```
workstation% tclsh
```

**Step 3** **pkg\_mkindex** *directory\_name* \*.tcl

Use the **pkg\_mkindex** command to create the pkgIndex file. The pkgIndex file contains a directory of all the packages contained in the Tcl library files. We recommend that you run **pkg\_mkindex** inside a directory because there can only be a single pkgIndex file in any directory and you may have other Tcl files to be grouped together. Running **pkg\_mkindex** in a directory determines which Tcl package file or files are indexed using a specific pkgIndex.

**Example:**

```
workstation% pkg_mkindex eem_library *.tcl
```

The following example pkgIndex is created when some Tcl package files are in a library file directory and the **pkg\_mkindex** command is run.

**pkgIndex**

**Example:**

```
Tcl package index file, version 1.1
This file is generated by the "pkg_mkIndex" command
and sourced either when an application starts up or
by a "package unknown" script. It invokes the
"package ifneeded" command to set up package-related
information so that packages will be loaded automatically
in response to "package require" commands. When this
script is sourced, the variable $dir must contain the
full path name of this file's directory.
package ifneeded xmlrpc 0.3 [list source [file join $dir xmlrpc.tcl]]
```

**Step 4** Copy the Tcl library files and the pkgIndex file to the directory used for storing user library files on the target device.

**Step 5** Copy a user-defined EEM policy file written in Tcl to the directory used for storing user-defined EEM policies on the target device.

The directory for storing user-defined EEM policies can be the same directory used in Step 4. The following example user-defined EEM policy can be used to test the Tcl package support in EEM.

**packagetest.tcl**

**Example:**

```
::cisco::eem::event_register_none maxrun 1000000.000
#
test if xmlrpc available
#
#
Namespace imports
```

```
#
namespace import ::cisco::eem::*
namespace import ::cisco::lib::*
#
package require xmlrpc
puts "Did you get an error?"
```

**Step 6**     **enable**

Enables privileged EXEC mode. Enter your password if prompted.

**Example:**

```
Device> enable
```

**Step 7**     **configure terminal**

Enables global configuration mode.

**Example:**

```
Device# configure terminal
```

**Step 8**     **event manager directory user library** *path*

Use this command to specify the EEM user library directory; this is the directory to which the files in were copied.

**Example:**

```
Device(config)# event manager directory user library disk2:/eem_library
```

**Step 9**     **event manager directory user policy** *path*

Use this command to specify the EEM user policy directory; this is the directory to which the file was copied.

**Example:**

```
Device(config)# event manager directory user policy disk2:/eem_policies
```

**Step 10**    **event manager policy** *policy-name* [**type** {system | user}] [**trap**]

Use this command to register a user-defined EEM policy. In this example, the policy named packetest.tcl is registered.

**Example:**

```
Device(config)# event manager policy packetest.tcl
```

**Step 11**    **event manager run** *policy-name*

Use this command to manually run an EEM policy. In this example, the policy named packetest.tcl is run to test the Tcl package support in EEM.

**Example:**

```
Device(config)# event manager run packetest.tcl
```



# Configuration Examples for Writing Embedded Event Manager Policies Using Tcl

## Assigning a Username for a Tcl Session Examples

The following example shows how to set a username to be associated with a Tcl session. If you are using authentication, authorization, and accounting (AAA) security and implement authorization on a command basis, you should use the **event manager session cli username** command to set a username to be associated with a Tcl session. The username is used when a Tcl policy executes a CLI command. TACACS+ verifies each CLI command using the username associated with the Tcl session that is running the policy. Commands from Tcl policies are not usually verified because the device must be in privileged EXEC mode to register the policy. In the example, the username is yourname, and this is the username that is used whenever a CLI command session is initiated from within an EEM policy.

```
configure terminal
event manager session cli username yourname
end
```

## EEM Event Detector Demo Examples

### EEM Sample Policy Descriptions

This configuration example features some of the sample EEM policies:

- `ap_perf_test_base_cpu.tcl`--Is run to measure the the CPU performance of EEM policies.
- `no_perf_test_init.tcl`--Is run to measure the CPU performance of EEM policies.
- `sl_intf_down.tcl`--Is run when a configurable syslog message is logged. It executes up to two configurable CLI commands and e-mails the results.
- `tm_cli_cmd.tcl`--Is run using a configurable CRON entry. It executes a configurable CLI command and e-mails the results.
- `tm_crash_reporter.tcl`--Is run 5 seconds after it is registered and 5 seconds after the device boots up. When triggered, the script attempts to find the reload reason. If the reload reason was due to a crash, the policy searches for the related crashinfo file and sends this information to a URL location specified by the user in the environment variable `_crash_reporter_url`.
- `tm_fsys_usage.tcl`--This policy runs using a configurable CRON entry and monitors disk space usage. A syslog message is displayed if disk space usage crosses configurable thresholds.

### Event Manager Environment Variables for the Sample Policies

Event manager environment variables are Tcl global variables that are defined external to the EEM policy before the policy is registered and run. The sample policies require three of the e-mail environment variables to be set ; only `_email_cc` is optional. Other required and optional variable settings are outlined in the following tables.

The table below describes the EEM environment variables that must be set before the `ap_perf_test_base_cpu.tcl` sample policy is run.

**Table 212: Environment Variables Used in the `ap_perf_test_base_cpu.tcl` Policy**

| Environment Variable          | Description                                                                                                                                                                                                    | Example                                                |
|-------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------|
| <code>_perf_iterations</code> | The number of iterations over which to run the measurement.                                                                                                                                                    | <b>100</b>                                             |
| <code>_perf_cmd1</code>       | The first non interactive CLI command that is executed as part of the measurement test. This variable is optional and need not be specified.                                                                   | <b>enable</b>                                          |
| <code>_perf_cmd2</code>       | The second non interactive CLI command that is as part of the measurement test. To use <code>_perf_cmd2</code> , <code>_perf_cmd1</code> must be defined. This variable is optional and need not be specified. | <b>show version</b>                                    |
| <code>_perf_cmd3</code>       | The third non interactive CLI command that is as part of the measurement test. To use <code>_perf_cmd3</code> , <code>_perf_cmd1</code> must be defined. This variable is optional and need not be specified.  | <b>show interface<br/>counters protocol<br/>status</b> |

The table below describes the EEM environment variables that must be set before the `no_perf_test_init.tcl` sample policy is run.

**Table 213: Environment Variables Used in the `no_perf_test_init.tcl` Policy**

| Environment Variable          | Description                                                                                                                                                                                                    | Example                                                |
|-------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------|
| <code>_perf_iterations</code> | The number of iterations over which to run the measurement.                                                                                                                                                    | <b>100</b>                                             |
| <code>_perf_cmd1</code>       | The first non interactive CLI command that is executed as part of the measurement test. This variable is optional and need not be specified.                                                                   | <b>enable</b>                                          |
| <code>_perf_cmd2</code>       | The second non interactive CLI command that is as part of the measurement test. To use <code>_perf_cmd2</code> , <code>_perf_cmd1</code> must be defined. This variable is optional and need not be specified. | <b>show version</b>                                    |
| <code>_perf_cmd3</code>       | The third non interactive CLI command that is as part of the measurement test. To use <code>_perf_cmd3</code> , <code>_perf_cmd1</code> must be defined. This variable is optional and need not be specified.  | <b>show interface<br/>counters protocol<br/>status</b> |

The table below describes the EEM environment variables that must be set before the `sl_intf_down.tcl` sample policy is run.

**Table 214: Environment Variables Used in the `sl_intf_down.tcl` Policy**

| Environment Variable      | Description                                       | Example                      |
|---------------------------|---------------------------------------------------|------------------------------|
| <code>_config_cmd1</code> | The first configuration command that is executed. | <b>interface Ethernet1/0</b> |

| Environment Variable | Description                                                                                                          | Example                      |
|----------------------|----------------------------------------------------------------------------------------------------------------------|------------------------------|
| _config_cmd2         | The second configuration command that is executed. This variable is optional and need not be specified.              | <b>no shutdown</b>           |
| _syslog_pattern      | A regular expression pattern match string that is used to compare syslog messages to determine when the policy runs. | . *UPDOWN.*FastEthernet0/0.* |

The table below describes the EEM environment variables that must be set before the tm\_cli\_cmd.tcl sample policy is run.

**Table 215: Environment Variables Used in the tm\_cli\_cmd.tcl Policy**

| Environment Variable | Description                                                    | Example               |
|----------------------|----------------------------------------------------------------|-----------------------|
| _cron_entry          | A CRON specification that determines when the policy will run. | 0-59/1 0-23/1 * * 0-7 |
| _show_cmd            | The CLI command to be executed when the policy is run.         | <b>show version</b>   |

The table below describes the EEM environment variables that must be set before the tm\_crash\_reporter.tcl sample policy is run.

**Table 216: Environment Variables Used in the tm\_crash\_reporter.tcl Policy**

| Environment Variable  | Description                                                                                                                                       | Example                                    |
|-----------------------|---------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------|
| _crash_reporter_debug | A value that identifies whether debug information for tm_crash_reporter.tcl will be enabled. This variable is optional and need not be specified. | 1                                          |
| _crash_reporter_url   | The URL location to which the crash report is sent.                                                                                               | http://www.example.com/fm/interface_tm.cgi |

The table below describes the EEM environment variables that must be set before the tm\_fsys\_usage.tcl sample policy is run.

**Table 217: Environment Variables Used in the tm\_fsys\_usage.tcl Policy**

| Environment Variable | Description                                                                                                                                                                                                           | Example               |
|----------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------|
| _tm_fsys_usage_cron  | A CRON specification that is used in the <b>event_register</b> Tcl command extension. If unspecified, the tm_fsys_usage.tcl policy is triggered once per minute. This variable is optional and need not be specified. | 0-59/1 0-23/1 * * 0-7 |
| _tm_fsys_usage_debug | When this variable is set to a value of 1, disk usage information is displayed for all entries in the system. This variable is optional and need not be specified.                                                    | 1                     |

| Environment Variable     | Description                                                                                                                                                                                                                                                                               | Example            |
|--------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------|
| _tm_fsys_usage_freebytes | Free byte threshold for systems or specific prefixes. If free space falls below a given value, a warning is displayed. This variable is optional and need not be specified.                                                                                                               | disk2:98000000     |
| _tm_fsys_usage_percent   | Disk usage percentage thresholds for systems or specific prefixes. If the disk usage percentage exceeds a given percentage, a warning is displayed. If unspecified, the default disk usage percentage is 80 percent for all systems. This variable is optional and need not be specified. | nvrnram:25 disk2:5 |

### Registration of Some EEM Policies

Some EEM policies must be unregistered and then reregistered if an EEM environment variable is modified after the policy is registered. The `event_register_xxx` statement that appears at the start of the policy contains some of the EEM environment variables, and this statement is used to establish the conditions under which the policy is run. If the environment variables are modified after the policy has been registered, the conditions may become invalid. To avoid any errors, the policy must be unregistered and then reregistered. The following variables are affected:

- `_cron_entry` in the `tm_cli_cmd.tcl` policy
- `_syslog_pattern` in the `sl_intf_down.tcl` policy

### Basic Configuration Details for All Sample Policies

To allow e-mail to be sent from the Embedded Event Manager, the `hostname` and `ip domain-name` commands must be configured. The EEM environment variables must also be set. After a Cisco IOS image has been booted, use the following initial configuration, substituting appropriate values for your network. The environment variables for the `tm_fsys_usage` sample policy (see the table above) are all optional and are not listed here:

```
hostname cpu
ip domain-name example.com
event manager environment _email_server ms.example.net
event manager environment _email_to username@example.net
event manager environment _email_from engineer@example.net
event manager environment _email_cc projectgroup@example.net
event manager environment _cron_entry 0-59/2 0-23/1 * * 0-7
event manager environment _show_cmd show event manager policy registered
event manager environment _syslog_pattern .*UPDOWN.*FastEthernet0/0
event manager environment _config_cmd1 interface Ethernet1/0
event manager environment _config_cmd2 no shutdown
event manager environment _crash_reporter_debug 1
event manager environment _crash_reporter_url
http://www.example.com/fm/interface_tm.cgi
end
```

### Using the Sample Policies

This section contains the following configuration scenarios to demonstrate how to use the some sample Tcl policies:

### Running the Mandatory.go\_\*.tcl Sample Policy

There are GOLD TCL scripts for each test which runs as a part of GOLD EEM Policy. You can modify the TCL script for the test, specify the consecutive failure count, and also change the default corrective action. For example, one could choose to power down a linecard card, instead of reset or other CLI based actions.

For each registered test, a default TCL script is available, which can be registered with the system, and matches with the default action. This can be then overridden by modifying these scripts.

The following table shows a list of the mandatory policies that GOLD installed into EEM. Each of the policies performs some sort of action such as resetting the card or disabling the port.

| GOLD Tcl Scripts            | Test                         |
|-----------------------------|------------------------------|
| Mandatory.go_asicsync.tcl   | TestAsicSync                 |
| Mandatory.go_bootup.tcl     | Common for all bootup tests. |
| Mandatory.go_fabric.tcl     | TestFabricHealth             |
| Mandatory.go_fabrich0.tcl   | TestFabricCh0Health          |
| Mandatory.go_fabrich1.tcl   | TestFabricCh1Health          |
| Mandatory.go_ipsec.tcl      | TestIPSecEncrypDecrypPkt     |
| Mandatory.go_mac.tcl        | TestMacNotification          |
| Mandatory.go_nondislp.tcl   | TestNonDisruptiveLoopback    |
| Mandatory.go_scratchreg.tcl | TestScratchRegister          |
| Mandatory.go_sprping.tcl    | TestSPRPIbandPing            |

The following sample configuration demonstrates how to use this policy. Starting in user EXEC mode, enter the **enable** command at the device prompt. The device enters privileged EXEC mode, where you can enter the **show event manager policy registered** command to verify that no policies are currently registered. The next command is the **show event manager policy available** command to display which policies are available to be installed. After you enter the **configure terminal** command to reach global configuration mode, you can register the mandatory.go\_\*.tcl policy with EEM using the **event manager policy** command. Exit from global configuration mode and enter the **show event manager policy registered** command again to verify that the policy has been registered.

```
enable
show event manager policy registered
show event manager policy available
configure terminal
 event manager policy Mandatory.go_spuriousisr.tcl
end
show event manager policy registered
show event manager environment
```

### Running the ap\_perf\_test\_base\_cpu.tcl and no\_perf\_test\_init.tcl Sample Policies

These sample policies measure the CPU performance of EEM policies. The policies help find the average execution time of each EEM policy and use the CLI library to execute the configuration commands specified in the EEM environment variables `_perf_cmd1` and, optionally, `_perf_cmd2` and `_perf_cmd3`.

The following sample configuration demonstrates how to use this policy. Starting in user EXEC mode, enter the **enable** command at the device prompt. The device enters privileged EXEC mode, where you can enter the **show event manager policy registered** command to verify that no policies are currently registered. The next command is the **show event manager policy available** command to display which policies are available to be installed. After you enter the **configure terminal** command to reach global configuration mode, enter the **service timestamps debug datetime msec** command and then you can register the `ap_perf_test_base_cpu.tcl` and `no_perf_test_init.tcl` policies with EEM using the **event manager policy** command. Exit from global configuration mode and enter the **show event manager policy registered** command again to verify that the policy has been registered.

The policies `ap_perf_test_base_cpu.tcl` and `no_perf_test_init.tcl` need to be registered together, as they run as a test suite. You can run the `no_perf_test_init.tcl` policy to start the tests. Analyze the results using the syslog messages from each iteration. The total number of iteration is specified by the variable `_perf_iterations`. Take the time difference and divide it by the total number of iterations to get the average execution time of each EEM policy.

```
enable
show event manager policy registered
show event manager policy available
show event manager environment
configure terminal
 service timestamps debug datetime msec
 event manager environment _perf_iterations 100
 event manager policy ap_perf_test_base_cpu.tcl
 event manager policy no_perf_test_init.tcl
end
show event manager policy registered
show event manager policy available
show event manager environment
event manager run no_perf_test_init.tcl
```

### Running the `no_perf_test_init.tcl` Sample Policy

This sample policy measures the the cpu performance of EEM policies. The policy helps to find the average execution time of each EEM policy and uses the CLI library to execute the configuration commands specified in the EEM environment variables `_perf_cmd1` and, optionally, `_perf_cmd2` and `_perf_cmd3`.

The following sample configuration demonstrates how to use this policy. Starting in user EXEC mode, enter the **enable** command at the device prompt. The device enters privileged EXEC mode, where you can enter the **show event manager policy registered** command to verify that no policies are currently registered. The next command is the **show event manager policy available** command to display which policies are available to be installed. After you enter the **configure terminal** command to reach global configuration mode, you can register the `no_perf_test_init.tcl` policy with EEM using the **event manager policy** command. Exit from global configuration mode and enter the **show event manager policy registered** command again to verify that the policy has been registered.

Analyze the results using the syslog messages from each iteration. The total number of iteration is specified by the variable `_perf_iterations`. Take the time difference and divide it by the total number of iterations to get the average execution time of each EEM policy.

```
enable
show event manager policy registered
show event manager policy available
configure terminal
 event manager policy no_perf_test_init.tcl
end
```

```
show event manager policy registered
show event manager environment
```

### Running the `sl_intf_down.tcl` Sample Policy

This sample policy demonstrates the ability to modify the configuration when a syslog message with a specific pattern is logged. The policy gathers detailed information about the event and uses the CLI library to execute the configuration commands specified in the EEM environment variables `_config_cmd1` and, optionally, `_config_cmd2`. An e-mail message is sent with the results of the CLI command.

The following sample configuration demonstrates how to use this policy. Starting in user EXEC mode, enter the **enable** command at the device prompt. The device enters privileged EXEC mode, where you can enter the **show event manager policy registered** command to verify that no policies are currently registered. The next command is the **show event manager policy available** command to display which policies are available to be installed. After you enter the **configure terminal** command to reach global configuration mode, you can register the `sl_intf_down.tcl` policy with EEM using the **event manager policy** command. Exit from global configuration mode and enter the **show event manager policy registered** command again to verify that the policy has been registered.

The policy runs when an interface goes down. Enter the **show event manager environment** command to display the current environment variable values. Unplug the cable (or configure a shutdown) for the interface specified in the `_syslog_pattern` EEM environment variable. The interface goes down, prompting the syslog daemon to log a syslog message about the interface being down, and the syslog event detector is called.

The syslog event detector reviews the outstanding event specifications and finds a match for interface status change. The EEM server is notified, and the server runs the policy that is registered to handle this event--`sl_intf_down.tcl`.

```
enable
show event manager policy registered
show event manager policy available
configure terminal
 event manager policy sl_intf_down.tcl
end
show event manager policy registered
show event manager environment
```

### Running the `tm_cli_cmd.tcl` Sample Policy

This sample policy demonstrates the ability to periodically execute a CLI command and to e-mail the results. The CRON specification “0-59/2 0-23/1 \* \* 0-7” causes this policy to be run on the second minute of each hour. The policy gathers detailed information about the event and uses the CLI library to execute the configuration commands specified in the EEM environment variable `_show_cmd`. An e-mail message is sent with the results of the CLI command.

The following sample configuration demonstrates how to use this policy. Starting in user EXEC mode, enter the **enable** command at the device prompt. The device enters privileged EXEC mode where you can enter the **show event manager policy registered** command to verify that no policies are currently registered. The next command is the **show event manager policy available** command to display which policies are available to be installed. After you enter the **configure terminal** command to reach global configuration mode, you can register the `tm_cli_cmd.tcl` policy with EEM using the **event manager policy** command. Exit from global configuration mode and enter the **show event manager policy registered** command to verify that the policy has been registered.

The timer event detector triggers an event for this case periodically according to the CRON string set in the EEM environment variable `_cron_entry`. The EEM server is notified, and the server runs the policy that is registered to handle this event--`tm_cli_cmd.tcl`.

```
enable
show event manager policy registered
show event manager policy available
configure terminal
 event manager policy tm_cli_cmd.tcl
end
show event manager policy registered
```

### Running the `tm_crash_reporter.tcl` Sample Policy

This sample policy demonstrates the ability to send an HTTP-formatted crash report to a URL location. If the policy registration is saved in the startup configuration file, the policy is triggered 5 seconds after bootup. When triggered, the script attempts to find the reload reason. If the reload reason was due to a crash, the policy searches for the related crashinfo file and sends this information to a URL location specified by the user in the environment variable `_crash_reporter_url`. A CGI script, `interface_tm.cgi`, has been created to receive the URL from the `tm_crash_reporter.tcl` policy and save the crash information in a local database on the target URL machine.

A Perl CGI script, `interface_tm.cgi`, has been created and is designed to run on a machine that contains an HTTP server and is accessible by the device that runs the `tm_crash_reporter.tcl` policy. The `interface_tm.cgi` script parses the data passed into it from `tm_crash_reporter.tcl` and appends the crash information to a text file, creating a history of all crashes in the system. Additionally, detailed information on each crash is stored in three files in a crash database directory that is specified by the user. Another Perl CGI script, `crash_report_display.cgi`, has been created to display the information stored in the database created by the `interface_tm.cgi` script. The `crash_report_display.cgi` script should be placed on the same machine that contains `interface_tm.cgi`. The machine should be running a web browser such as Internet Explorer or Netscape. When the `crash_report_display.cgi` script is run, it displays the crash information in a readable format.

The following sample configuration demonstrates how to use this policy. Starting in user EXEC mode, enter the **enable** command at the device prompt. The device enters privileged EXEC mode where you can enter the **show event manager policy registered** command to verify that no policies are currently registered. The next command is the **show event manager policy available** command to display which policies are available to be installed. After you enter the **configure terminal** command to reach global configuration mode, you can register the `tm_crash_reporter.tcl` policy with EEM using the **event manager policy** command. Exit from global configuration mode and enter the **show event manager policy registered** command to verify that the policy has been registered.

```
enable
show event manager policy registered
show event manager policy available
configure terminal
 event manager policy tm_crash_reporter.tcl
end
show event manager policy registered
```

### Running the `tm_fsys_usage.tcl` Sample Policy

This sample policy demonstrates the ability to periodically monitor disk space usage and report through syslog when configurable thresholds have been crossed.



The following sample configuration demonstrates how to use this policy. Starting in user EXEC mode, enter the **enable** command at the device prompt. The device enters privileged EXEC mode, where you can enter the **show event manager policy registered** command to verify that no policies are currently registered. The next command is the **show event manager policy available** command to display which policies are available to be installed. After you enter the **configure terminal** command to reach global configuration mode, you can register the `tm_fsys_usage.tcl` policy with EEM using the **event manager policy** command. Exit from global configuration mode and enter the **show event manager policy registered** command again to verify that the policy has been registered. If you had configured any of the optional environment variables that are used in the `tm_fsys_usage.tcl` policy, the **show event manager environment** command displays the configured variables.

```
enable
show event manager policy registered
show event manager policy available
configure terminal
 event manager policy tm_fsys_usage.tcl
 end
show event manager policy registered
show event manager environment
```

## Programming Policies with Tcl Sample Scripts Example

This section contains some of the sample policies that are included as EEM system policies. For more details about these policies, see the [EEM Event Detector Demo Examples, on page 2111](#).

### Mandatory.go\_ipsec.tcl Sample Policy

The following sample policy for the TestIPSecEncrypDecrypPkt Test.

```
::cisco::eem::event_register_gold card all testing_type monitoring test_name TestIPSecEncrypDecrypPkt consecutive_failure 6 platform_action 0 queue_priority last
#
GOLD TestIPSecEncrypDecrypPkt Test TCL script
#
March 2005, Hai Qiu
#
Copyright (c) 2005-2007 by cisco Systems, Inc.
All rights reserved.
#
#
Register for TestIPSecEncrypDecrypPkt test even
the elements for register the event
card [all | card #]
sub_card [all | sub_card #]
severity_major | severity_minor | severity_normal default : severity_normal
new_failure [true | false] default: dont_care
testing_type [bootup | ondemand | schedule | monitoring]
test_name [test name]
test_id [test #]
consecutive_failure [consecutive_failure #]
platform_action [action_flag]
action_flag [0 | 1 | 2]
queue_priority [normal | low | high | last] default: normal
#
Note:
1: "card" element is required. If other elements are not specified,
treat them as dont care, or default.
```

```

#
2: action_flag is platform specific. It is up to platform to
determine what action need to be taken based on the value
For Cat6k platform
action_flag 0 : TCL script take action to reset card
action_flag 1 : TCL script doesn't take action to reset card
action_flag 2 : TCL script takes action to reset card for bootup diag
when there is major error
action_flag 3 : TCL script doesn't take action to reset card for
bootup diag when there is major error
#
3: "queue_priority last" would guarantee this policy will be executed last
if there are other EEM events in queue with queue priority other
than "last"
namespace import ::cisco::eem::*
namespace import ::cisco::lib::*
1. query the information of latest triggered eem event
array set arr_einfo [event_reqinfo]
if {$_cerrno != 0} {
 set result [format "component=%s; subsys err=%s; posix err=%s;\n%s" \
 $_cerr_sub_num $_cerr_sub_err $_cerr_posix_err $_cerr_str]
 error $result
}
puts "GOLD EEM TCL policy for TestIPSecEncrypDecrypPkt"
#set msg [format "array=%s", array names arr_einfo]
#puts "msg $msg"
#set msg $arr_einfo(msg)
set card $arr_einfo(card)
set sub_card $arr_einfo(sub_card)
#set overall_result $arr_einfo(overall_result)
#puts "GOLD event msg recieved: $card/$sub_card overall_result= $overall_result"
2. execute the user-defined config commands
if [catch {cli_open} result] {
 error $result $errorInfo
} else {
 array set cli1 $result
}
if [catch {cli_exec $cli1(fd) "en"} result] {
 error $result $errorInfo
}
Use "diagn action mod mod# test testname default" command
for default platform action
if [catch {cli_exec $cli1(fd) "diagnostic action mod $card test TestIPSecEncrypD
ecrypPkt default"} result] {
 error $result $errorInfo
} else {
 set cmd_output $result
}
if [catch {cli_close $cli1(fd) $cli1(tty_id)} result] {
 error $result $errorInfo
}
}

```

### ap\_perf\_test\_base\_cpu.tcl Sample Policy

The following sample policy measures the CPU performance of EEM policies.

```

::cisco::eem::event_register_appl sub_system 798 type 9999
#-----
EEM policy used for measuring the cpu performance of EEM policies.
#
July 2005, Cisco EEM team
#
Copyright (c) 2005, 2006 by cisco Systems, Inc.

```

```

All rights reserved.
#-----
###
Input arguments:
###
arg1 $iter - current iteration count
###
The following EEM environment variables are used:
###
_perf_iterations (mandatory) - number of iterations over which we
will run our measurement.
Example:
event manager environment _perf_iterations 100
###
_perf_cmd1 (optional) - optional non interactive cli command
to be executed as part of the
measurement test.
Example:
event manager environment _perf_cmd1 enable
###
_perf_cmd2 (optional) - optional non interactive cli command
to be executed as part of the
measurement test.
To use _perf_cmd2, _perf_cmd1 MUST
be defined.
Example:
event manager environment _perf_cmd2 show ver
###
_perf_cmd3 (optional) - optional non interactive cli command
to be executed as part of the
measurement test.
To use _perf_cmd3, _perf_cmd1 MUST
be defined.
Example:
event manager environment _perf_cmd3 show int counters protocol status
###
Description:
Iterate through _perf_iterations of this policy.
It is up to the user to calculate the average
execution time based on the system timestamps.
Optional commands _perf_cmd1,
_perf_cmd2 and _perf_cmd3 are executed if defined.
###
A value of 100 is a good starting point.
###
Outputs:
Console output.
###
Usage example:
>conf t
>service timestamps debug datetime msec
>event manager environment _perf_iterations 100
>event manager policy ap_perf_base_cpu.tcl
>event manager policy no_perf_test_init.tcl
>end
2d19h: %SYS-5-CONFIG_I: Configured from console by console
>event manager run no_perf_test_init.tcl
###
Oct 16 14:57:17.284: %SYS-5-CONFIG_I: Configured from console by console
>event manager run no_perf_test_init.tcl
###
Oct 16 19:32:02.772: %HA_EM-6-LOG:
eem_policy/no_perf_test_init.tcl: EEM performance test start
Oct 16 19:32:03.115: %HA_EM-6-LOG:

```

```

eem_policy/ap_perf_test_base_cpu.tcl: EEM performance test iteration 1
Oct 16 19:32:03.467: %HA_EM-6-LOG:
eem_policy/ap_perf_test_base_cpu.tcl: EEM performance test iteration 2
...
Oct 16 19:32:36.936: %HA_EM-6-LOG:
eem_policy/ap_perf_test_base_cpu.tcl: EEM performance test iteration 100
Oct 16 19:32:36.936: %HA_EM-6-LOG:
eem_policy/ap_perf_test_base_cpu.tcl: EEM performance test end
###
The user must calculate execution time and average time of execution.
In this example, total time = 19:32:36.936 - 19:32:02.772 = 34.164
Average script execution time = 341.64 milliseconds
###
check if all the env variables we need exist
If any of them doesn't exist, print out an error msg and quit
if ![info exists _perf_iterations] {
 set result \
 "Policy cannot be run: variable _perf_iterations has not been set"
 error $result $errorInfo
}
ensure our target iteration count > 0
if {$_perf_iterations <= 0} {
 set result \
 "Policy cannot be run: variable _perf_iterations <= 0"
 error $result $errorInfo
}
namespace import ::cisco::eem::*
namespace import ::cisco::lib::*
query the event info
array set arr_einfo [event_reqinfo]
if {$_cerrno != 0} {
 set result [format "component=%s; subsys err=%s; posix err=%s;\n%s" \
 $_cerr_sub_num $_cerr_sub_err $_cerr_posix_err $_cerr_str]
 error $result
}
set iter $arr_einfo(data1)
set iter [expr $iter + 1]
if _perf_cmd1 is defined
if {[info exists _perf_cmd1]} {
 # open the cli library
 if [catch {cli_open} result] {
 error $result $errorInfo
 } else {
 array set cli1 $result
 }
 # execute the comamnd defined in _perf_cmd1
 if [catch {cli_exec $cli1(fd) $_perf_cmd1} result] {
 error $result $errorInfo
 }
 # if _perf_cmd2 is defined
 if {[info exists _perf_cmd2]} {
 # execute the comamnd defined in _perf_cmd2
 if [catch {cli_exec $cli1(fd) $_perf_cmd2} result] {
 error $result $errorInfo
 } else {
 set cmd_output $result
 }
 }
 # if _perf_cmd3 is defined
 if {[info exists _perf_cmd3]} {
 # execute the comamnd defined in _perf_cmd3
 if [catch {cli_exec $cli1(fd) $_perf_cmd3} result] {
 error $result $errorInfo
 } else {

```

```

 set cmd_output $result
 }
}
close the cli library
if [catch {cli_close $cli1(fd) $cli1(tty_id)} result] {
 error $result $errorInfo
}
}

log a message
set msg [format "EEM performance test iteration %s" $iter]
action_syslog priority info msg $msg
if {$_cerrno != 0} {
 set result [format "component=%s; subsys err=%s; posix err=%s;\n%s" \
 $_cerr_sub_num $_cerr_sub_err $_cerr_posix_err $_cerr_str]
 error $result
}
use the context info from the previous run to determine when to end
if {$iter >= $_perf_iterations} {
 #log the final messages
 action_syslog priority info msg "EEM performance test end"
 if {$_cerrno != 0} {
 set result [format \
 "component=%s; subsys err=%s; posix err=%s;\n%s" \
 $_cerr_sub_num $_cerr_sub_err $_cerr_posix_err $_cerr_str]
 error $result
 }
 exit 0
}
cause the next iteration to run
event_publish sub_system 798 type 9999 arg1 $iter
if {$_cerrno != 0} {
 set result [format \
 "component=%s; subsys err=%s; posix err=%s;\n%s" \
 $_cerr_sub_num $_cerr_sub_err $_cerr_posix_err $_cerr_str]
 error $result
}
}

```

### tm\_cli\_cmd.tcl Sample Policy

The following sample policy runs a configurable CRON entry. The policy executes a configurable Cisco IOS CLI command and e-mails the results. An optional log file can be defined to which the output is appended with a timestamp.

```

::cisco::eem::event_register_timer cron name crontimer2 cron_entry $_cron_entry maxrun 240
#-----
EEM policy that will periodically execute a cli command and email the
results to a user.
#
July 2005, Cisco EEM team
#
Copyright (c) 2005 by cisco Systems, Inc.
All rights reserved.
#-----
The following EEM environment variables are used:
###
_cron_entry (mandatory) - A CRON specification that determines
when the policy will run. See the
IOS Embedded Event Manager
documentation for more information
on how to specify a cron entry.

```

```

Example: _cron_entry 0-59/1 0-23/1 * * 0-7
###
_log_file (mandatory without _email_....)
- A filename to append the output to.
If this variable is defined, the
output is appended to the specified
file with a timestamp added.
Example: _log_file bootflash:/my_file.log
###
_email_server (mandatory without _log_file)
- A Simple Mail Transfer Protocol (SMTP)
mail server used to send e-mail.
Example: _email_server mailserver.example.com
###
_email_from (mandatory without _log_file)
- The address from which e-mail is sent.
Example: _email_from devtest@example.com
###
_email_to (mandatory without _log_file)
- The address to which e-mail is sent.
Example: _email_to engineering@example.com
###
_email_cc (optional)
- The address to which the e-mail must
be copied.
Example: _email_cc manager@example.com
###
_show_cmd (mandatory)
- The CLI command to be executed when
the policy is run.
Example: _show_cmd show version
###
check if all required environment variables exist
If any required environment variable does not exist, print out an error msg and quit
if {[info exists _log_file]} {
 if {[info exists _email_server]} {
 set result \
 "Policy cannot be run: variable _log_file or _email_server has not been set"
 error $result $errorMsg
 }
 if {[info exists _email_from]} {
 set result \
 "Policy cannot be run: variable _log_file or _email_from has not been set"
 error $result $errorMsg
 }
 if {[info exists _email_to]} {
 set result \
 "Policy cannot be run: variable _log_file ore _email_to has not been set"
 error $result $errorMsg
 }
 if {[info exists _email_cc]} {
 #_email_cc is an option, must set to empty string if not set.
 set _email_cc ""
 }
}
if {[info exists _show_cmd]} {
 set result \
 "Policy cannot be run: variable _show_cmd has not been set"
 error $result $errorMsg
}
namespace import ::cisco::eem::*
namespace import ::cisco::lib::*
query the event info and log a message
array set arr_einfo [event_reginfo]
if {$_cerrno != 0} {
 set result [format "component=%s; subsys err=%s; posix err=%s;\n%s" \

```

```

 $_cerr_sub_num $_cerr_sub_err $_cerr_posix_err $_cerr_str]
 error $result
}
global timer_type timer_time_sec
set timer_type $arr_einfo(timer_type)
set timer_time_sec $arr_einfo(timer_time_sec)
log a message
set msg [format "timer event: timer type %s, time expired %s" \
 $timer_type [clock format $timer_time_sec]]
action_syslog priority info msg $msg
if {$_cerrno != 0} {
 set result [format "component=%s; subsys err=%s; posix err=%s;\n%s" \
 $_cerr_sub_num $_cerr_sub_err $_cerr_posix_err $_cerr_str]
 error $result
}
1. execute the command
if [catch {cli_open} result] {
 error $result $errorInfo
} else {
 array set cli1 $result
}
if [catch {cli_exec $cli1(fd) "en"} result] {
 error $result $errorInfo
}
save exact execution time for command
set time_now [clock seconds]
execute command
if [catch {cli_exec $cli1(fd) $_show_cmd} result] {
 error $result $errorInfo
} else {
 set cmd_output $result
 # format output: remove trailing router prompt
 regexp {\n*(.*\n)([^\n]*)$} $result dummy cmd_output
}
if [catch {cli_close $cli1(fd) $cli1(tty_id)} result] {
 error $result $errorInfo
}

2. log the success of the CLI command
set msg [format "Command \"%s\" executed successfully" $_show_cmd]
action_syslog priority info msg $msg
if {$_cerrno != 0} {
 set result [format "component=%s; subsys err=%s; posix err=%s;\n%s" \
 $_cerr_sub_num $_cerr_sub_err $_cerr_posix_err $_cerr_str]
 error $result
}
3. if _log_file is defined, then attach it to the file
if {[info exists _log_file]} {
 # attach output to file
 if [catch {open $_log_file a+} result] {
 error $result
 }
 set fileD $result
 # save timestamp of command execution
 # (Format = 00:53:44 PDT Mon May 02 2005)
 set time_now [clock format $time_now -format "%T %Z %a %b %d %Y"]
 puts $fileD "%%% Timestamp = $time_now"
 puts $fileD $cmd_output
 close $fileD
}
4. if _email_server is defined send the email out
if {[info exists _email_server]} {
 set routename [info hostname]
 if {[string match "" $routename]} {

```

```

error "Host name is not configured"
}
if [catch {smtp_subst [file join $tcl_library email_template_cmd.tm]} \
result] {
error $result $ErrorInfo
}
if [catch {smtp_send_email $result} result] {
error $result $ErrorInfo
}
}
}

```

### sl\_intf\_down.tcl Sample Policy

The following sample policy runs when a configurable syslog message is logged. The policy executes a configurable CLI command and e-mails the results.

```

::cisco::eem::event_register_syslog occurs 1 pattern $_syslog_pattern maxrun 90

#-----
EEM policy to monitor for a specified syslog message.
Designed to be used for syslog interface-down messages.
When event is triggered, the given config commands will be run.
#
July 2005, Cisco EEM team
#
Copyright (c) 2005 by cisco Systems, Inc.
All rights reserved.
#-----

The following EEM environment variables are used:
###
_syslog_pattern (mandatory) - A regular expression pattern match string
that is used to compare syslog messages
to determine when policy runs
Example: _syslog_pattern .*UPDOWN.*FastEthernet0/0.*
###
_email_server (mandatory) - A Simple Mail Transfer Protocol (SMTP)
mail server used to send e-mail.
Example: _email_server mailserver.example.com
###
_email_from (mandatory) - The address from which e-mail is sent.
Example: _email_from devtest@example.com
###
_email_to (mandatory) - The address to which e-mail is sent.
Example: _email_to engineering@example.com
###
_email_cc (optional) - The address to which the e-mail must
be copied.
Example: _email_cc manager@example.com
###
_config_cmd1 (optional) - The first configuration command that
is executed.
Example: _config_cmd1 interface Ethernet1/0
###
_config_cmd2 (optional) - The second configuration command that
is executed.
Example: _config_cmd2 no shutdown
###

check if all the env variables we need exist
If any of them doesn't exist, print out an error msg and quit
if {[info exists _email_server]} {
 set result \

```



```

 "Policy cannot be run: variable _email_server has not been set"
 error $result $errorInfo
 }
 if {[info exists _email_from]} {
 set result \
 "Policy cannot be run: variable _email_from has not been set"
 error $result $errorInfo
 }
 if {[info exists _email_to]} {
 set result \
 "Policy cannot be run: variable _email_to has not been set"
 error $result $errorInfo
 }
 if {[info exists _email_cc]} {
 #_email_cc is an option, must set to empty string if not set.
 set _email_cc ""
 }
}

namespace import ::cisco::eem::*
namespace import ::cisco::lib::*

1. query the information of latest triggered eem event
array set arr_einfo [event_reqinfo]

if {$_cerrno != 0} {
 set result [format "component=%s; subsys err=%s; posix err=%s;\n%s" \
 $_cerr_sub_num $_cerr_sub_err $_cerr_posix_err $_cerr_str]
 error $result
}

set msg $arr_einfo(msg)
set config_cmds ""

2. execute the user-defined config commands
if [catch {cli_open} result] {
 error $result $errorInfo
} else {
 array set cli1 $result
}
if [catch {cli_exec $cli1(fd) "en"} result] {
 error $result $errorInfo
}
if [catch {cli_exec $cli1(fd) "config t"} result] {
 error $result $errorInfo
}

if {[info exists _config_cmd1]} {
 if [catch {cli_exec $cli1(fd) $_config_cmd1} result] {
 error $result $errorInfo
 }
 append config_cmds $_config_cmd1
}

if {[info exists _config_cmd2]} {
 if [catch {cli_exec $cli1(fd) $_config_cmd2} result] {
 error $result $errorInfo
 }
 append config_cmds "\n"
 append config_cmds $_config_cmd2
}

if [catch {cli_exec $cli1(fd) "end"} result] {
 error $result $errorInfo
}
}

```

```

if [catch {cli_close $cli(fd) $cli(tty_id)} result] {
 error $result $errorInfo
}

after 60000
3. send the notification email
set routername [info hostname]
if {[string match "" $routername]} {
 error "Host name is not configured"
}

if [catch {smtp_subst [file join $tcl_library email_template_cfg.tm]} result] {
 error $result $errorInfo
}

if [catch {smtp_send_email $result} result] {
 error $result $errorInfo
}

```

The following e-mail template file is used with the EEM sample policy above:

```

email_template_cfg.tm
Mailservername: $_email_server
From: $_email_from
To: $_email_to
Cc: $_email_cc
Subject: From router $routername: Periodic $_show_cmd Output
$cmd_output

```

## Debugging Embedded Event Manager Policies Examples

The following examples show how to debug the CLI library and the SMTP library.

### Debugging the CLI Library

The CLI library allows users to run CLI commands and obtain the output of commands in Tcl. An Embedded Event Manager **debug** command has been provided for users of this library. The command to enable CLI library debugging is **debug event manager tcl cli\_library**. When enabled, this command displays all data that is passed in and read back from the TTY session that handles the CLI interactions. This data helps ensure users that the commands that they are passing to the CLI are valid.

### Example of the debug event manager tcl cli\_library Command

This example uses the sample policy `sl_intf_down.tcl`. When triggered, `sl_intf_down.tcl` passes a configuration command to the CLI through the CLI library. The command passed in below is **show event manager environment**. This command is not a valid command in configuration mode. Without the **debug** command enabled, the output is shown below:

```

00:00:57:sl_intf_down.tcl[0]:config_cmds are show eve man env
00:00:57:%SYS-5-CONFIG_I:Configured from console by vty0

```

Notice that with the output above the user would not know whether or not the command succeeded in the CLI. With the **debug event manager tcl cli\_library** command enabled, the user sees the following:

```

01:17:07: sl_intf_down.tcl[0]: DEBUG(cli_lib) : CTL : cli_open called.
01:17:07: sl_intf_down.tcl[0]: DEBUG(cli_lib) : OUT : nelson>
01:17:07: sl_intf_down.tcl[0]: DEBUG(cli_lib) : IN : nelson>enable
01:17:07: sl_intf_down.tcl[0]: DEBUG(cli_lib) : OUT : nelson#

```

```

01:17:07: sl_intf_down.tcl[0]: DEBUG(cli_lib) : IN : nelson#configure terminal
01:17:07: sl_intf_down.tcl[0]: DEBUG(cli_lib) : OUT : Enter configuration commands, one
per line. End with CNTL/Z.
01:17:07: sl_intf_down.tcl[0]: DEBUG(cli_lib) : OUT : nelson(config)#
01:17:07: sl_intf_down.tcl[0]: DEBUG(cli_lib) : IN : nelson(config)#show event manager
environment
01:17:07: sl_intf_down.tcl[0]: DEBUG(cli_lib) : OUT :
01:17:07: sl_intf_down.tcl[0]: DEBUG(cli_lib) : OUT : % Invalid input detected at '^'
marker.
01:17:07: sl_intf_down.tcl[0]: DEBUG(cli_lib) : OUT : nelson(config)#
01:17:07: sl_intf_down.tcl[0]: DEBUG(cli_lib) : IN : nelson(config)#end
01:17:07: sl_intf_down.tcl[0]: DEBUG(cli_lib) : OUT : nelson#
01:17:07: sl_intf_down.tcl[0]: DEBUG(cli_lib) : CTL : cli_close called.
01:17:07: sl_intf_down.tcl[0]: DEBUG(cli_lib) : IN : nelson#exit
01:17:07: sl_intf_down.tcl[0]: config_cmds are show event manager environment
01:17:07: %SYS-5-CONFIG_I: Configured from console by vty0

```

The output above shows that **show event manager environment** is an invalid command in configuration mode. The IN keyword signifies all data passed in to the TTY through the CLI library. The OUT keyword signifies all data read back from the TTY through the CLI library. The CTL keyword signifies helper functions used in the CLI library. These helper functions are used to set up and remove connections to the CLI.

### Debugging the SMTP Library

The SMTP library allows users to send e-mail messages to an SMTP e-mail server. An Embedded Event Manager **debug** command has been provided for users of this library. The command to enable SMTP library debugging is **debug event manager tcl smtp\_library**. When enabled, this command displays all data that is passed in and read back from the SMTP library routines. This data helps ensure users that the commands that they are passing to the SMTP library are valid.

### Example of the debug event manager tcl smtp\_library Command

This example uses the sample policy `tm_cli_cmd.tcl`. When triggered, `tm_cli_cmd.tcl` runs the command **show event manager policy available system** through the CLI library. The result is then mailed to a user through the SMTP library. The output will help debug any issues related to using the SMTP library.

With the **debug event manager tcl smtp\_library** command enabled, the users see the following on the console:

```

00:39:46: tm_cli_cmd.tcl[0]: DEBUG(smtp_lib) : smtp_read : 220 XXXX.example.com ESMTP XXXX
1.1.0; Tue,
25 Jun 2002 14:20:39 -0700 (PDT)
00:39:46: tm_cli_cmd.tcl[0]: DEBUG(smtp_lib) : smtp_write : HELO XXXX.example.com
00:39:46: tm_cli_cmd.tcl[0]: DEBUG(smtp_lib) : smtp_read : 250 XXXX.example.com Hello
XXXX.example.com [XXXX],
pleased to meet you
00:39:46: tm_cli_cmd.tcl[0]: DEBUG(smtp_lib) : smtp_write : MAIL FROM:<XX@example.com>
00:39:46: tm_cli_cmd.tcl[0]: DEBUG(smtp_lib) : smtp_read : 250 <XX@example.com>... Sender
ok
00:39:46: tm_cli_cmd.tcl[0]: DEBUG(smtp_lib) : smtp_write : RCPT TO:<XX@example.com>
00:39:47: tm_cli_cmd.tcl[0]: DEBUG(smtp_lib) : smtp_read : 250 <XX@example.com>... Recipient
ok
00:39:47: tm_cli_cmd.tcl[0]: DEBUG(smtp_lib) : smtp_write : RCPT TO:<XX@example.com>
00:39:47: tm_cli_cmd.tcl[0]: DEBUG(smtp_lib) : smtp_read : 250 <XX@example.com>... Recipient
ok
00:39:47: tm_cli_cmd.tcl[0]: DEBUG(smtp_lib) : smtp_write : DATA
00:39:47: tm_cli_cmd.tcl[0]: DEBUG(smtp_lib) : smtp_read : 354 Enter mail, end with "."
on a line by itself
00:39:47: tm_cli_cmd.tcl[0]: DEBUG(smtp_lib) : smtp_write : Date: 25 Jun 2002 14:35:00 UTC

```

```

00:39:47: tm_cli_cmd.tcl[0]: DEBUG(smtp_lib) : smtp_write : Message-ID:
<20020625143500.2387058729877@XXXX.example.com>
00:39:47: tm_cli_cmd.tcl[0]: DEBUG(smtp_lib) : smtp_write : From: XX@example.com
00:39:47: tm_cli_cmd.tcl[0]: DEBUG(smtp_lib) : smtp_write : To: XX@example.com
00:39:47: tm_cli_cmd.tcl[0]: DEBUG(smtp_lib) : smtp_write : Cc: XX@example.com
00:39:47: tm_cli_cmd.tcl[0]: DEBUG(smtp_lib) : smtp_write : Subject: From router nelson:
Periodic show eve man po ava system Output
00:39:47: tm_cli_cmd.tcl[0]: DEBUG(smtp_lib) : smtp_write : No. Type Time Created
Name
00:39:47: tm_cli_cmd.tcl[0]: DEBUG(smtp_lib) : smtp_write : 1 system Fri May3 20:42:34
2002 pr_cdp_abort.tcl
00:39:47: tm_cli_cmd.tcl[0]: DEBUG(smtp_lib) : smtp_write : 2 system Fri May3 20:42:54
2002 pr_iprouting_abort.tcl
00:39:47: tm_cli_cmd.tcl[0]: DEBUG(smtp_lib) : smtp_write : 3 system Wed Apr3 02:16:33
2002 sl_intf_down.tcl
00:39:47: tm_cli_cmd.tcl[0]: DEBUG(smtp_lib) : smtp_write : 4 system Mon Jun24 23:34:16
2002 tm_cli_cmd.tcl
00:39:47: tm_cli_cmd.tcl[0]: DEBUG(smtp_lib) : smtp_write : 5 system Wed Mar27 05:53:15
2002 tm_crash_hist.tcl
00:39:47: tm_cli_cmd.tcl[0]: DEBUG(smtp_lib) : smtp_write : nelson#
00:39:47: tm_cli_cmd.tcl[0]: DEBUG(smtp_lib) : smtp_write :
00:39:47: tm_cli_cmd.tcl[0]: DEBUG(smtp_lib) : smtp_write : .
00:39:47: tm_cli_cmd.tcl[0]: DEBUG(smtp_lib) : smtp_read : 250 ADE90179 Message accepted
for delivery
00:39:47: tm_cli_cmd.tcl[0]: DEBUG(smtp_lib) : smtp_write : QUIT
00:39:47: tm_cli_cmd.tcl[0]: DEBUG(smtp_lib) : smtp_read : 221 XXXX.example.com closing
connection

```

## Tracing Tcl set Command Operations Example

Tcl is a flexible language. One of the flexible aspects of Tcl is that you can override commands. In this example, the Tcl `set` command is renamed as `_set` and a new version of the `set` command is created that displays a message containing the text “setting” and appends the scalar variable that is being set. This example can be used to trace all instances of scalar variables being set.

```

rename set _set
proc set {var args} {
 puts [list setting $var $args]
 uplevel _set $var $args
};

```

When this is placed in a policy, a message is displayed anytime a scalar variable is set, for example:

```
02:17:58: sl_intf_down.tcl[0]: setting test_var 1
```

## RPC Event Detector Example

```

TCL script (rpccli.tcl):
::cisco::eem::event_register_rpc
namespace import ::cisco::eem::*
namespace import ::cisco::lib::*
proc run_cli { clist } {
 set rbuf ""
 if {[llength $clist] < 1} {
 return -code ok $rbuf
 }
 if {[catch {cli_open} result]} {
 return -code error $result
 }
}

```

```

 } else {
 array set cliarr $result
 }
 if {[catch {cli_exec $cliarr(fd) "enable"} result]} {
 return -code error $result
 }
 if {[catch {cli_exec $cliarr(fd) "term length 0"} result]} {
 return -code error $result
 }
 foreach cmd $clist {
 if {[catch {cli_exec $cliarr(fd) $cmd} result]} {
 return -code error $result
 }
 }
 append rbuf $result
 }
 if {[catch {cli_close $cliarr(fd) $cliarr(tty_id)} result]} {
 puts "WARNING: $result"
 }
 return -code ok $rbuf
}
}
proc run_cli_interactive { clist } {
 set rbuf ""
 if {[length $clist] < 1} {
 return -code ok $rbuf
 }
 if {[catch {cli_open} result]} {
 return -code error $result
 }
 } else {
 array set cliarr $result
 }
 if {[catch {cli_exec $cliarr(fd) "enable"} result]} {
 return -code error $result
 }
 if {[catch {cli_exec $cliarr(fd) "term length 0"} result]} {
 return -code error $result
 }
 }
 foreach cmd $clist {
 array set sendexp $cmd
 if {[catch {cli_write $cliarr(fd) $sendexp(send)} result]} {
 return -code error $result
 }
 }
 foreach response $sendexp(responses) {
 array set resp $response
 if {[catch {cli_read_pattern $cliarr(fd) $resp(expect)} result]} {
 return -code error $result
 }
 if {[catch {cli_write $cliarr(fd) $resp(reply)} result]} {
 return -code error $result
 }
 }
 if {[catch {cli_read $cliarr(fd)} result]} {
 return -code error $result
 }
 }
 append rbuf $result
 }
 if {[catch {cli_close $cliarr(fd) $cliarr(tty_id)} result]} {
 puts "WARNING: $result"
 }
 }
 return -code ok $rbuf
}
}
array set arr_einfo [event_reqinfo]
set args $arr_einfo(argc)
set cmds [list]
for { set i 0 } { $i < $args } { incr i } {

```

```

set arg "arg${i}"
Split each argument on the '^' character. The first element is
the command, and each subsequent element is a prompt followed by
a response to that prompt.
set cmdlist [split $arr_einfo($arg) "^"]
set cmdarr(send) [lindex $cmdlist 0]
set cmdarr(responses) [list]
if { [expr ([llength $cmdlist] - 1) % 2] != 0 } {
return -code 88
}
set cmdarr(responses) [list]
for { set j 1 } { $j < [llength $cmdlist] } { incr j 2 } {
set resp(expect) [lindex $cmdlist $j]
set resp(reply) [lindex $cmdlist [expr $j + 1]]
lappend cmdarr(responses) [array get resp]
}
lappend cmds [array get cmdarr]
}
set rc [catch {run_cli_interactive $cmds} output]
if { $rc != 0 } {
error $output $errorInfo
return -code 88
}
puts $output

```

## Additional References

The following sections provide references related to writing Embedded Event Manager policies using Tcl.

### Related Documents

| Related Topic                                                                                                  | Document Title                                                         |
|----------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------|
| EEM commands: complete command syntax, defaults, command mode, command history, usage guidelines, and examples | <a href="#">Cisco IOS Embedded Event Manager Command Reference</a>     |
| Embedded Event Manager overview                                                                                | Embedded Event Manager Overview module.                                |
| Embedded Event Manager policy writing using the CLI                                                            | Writing Embedded Event Manager Policies Using the Cisco IOS CLI module |
| Embedded Resource Manager                                                                                      | Embedded Resource Manager module                                       |

### MIBs

| MIB                          | MIBs Link                                                                                                                                                                                                              |
|------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| CISCO-EMBEDDED-EVENT-MGR-MIB | To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL:<br><a href="http://www.cisco.com/go/mibs">http://www.cisco.com/go/mibs</a> |

**RFCs**

| RFC                                                                                                                         | Title |
|-----------------------------------------------------------------------------------------------------------------------------|-------|
| No new or modified RFCs are supported by this feature, and support for existing RFCs has not been modified by this feature. | --    |

**Technical Assistance**

| Description                                                                                                                                                                                                                                                                                                                                                                           | Link                                                                                                              |
|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------|
| The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password. | <a href="http://www.cisco.com/cisco/web/support/index.html">http://www.cisco.com/cisco/web/support/index.html</a> |







## CHAPTER 102

# Signed Tcl Scripts

---

The Signed Tcl Scripts feature allows you to create a certificate to generate a digital signature and sign a Tool Command Language (Tcl) script with that digital signature. This feature also allows you to work with existing scripts and certificates. The digital signature is verified for authentication and then run with trusted access to the Tcl interpreter. If the script does not contain the digital signature, the script may run in a limited mode for untrusted scripts, or may not run at all.

- [Prerequisites for Signed Tcl Scripts, on page 2135](#)
- [Restrictions for Signed Tcl Scripts, on page 2135](#)
- [Information About Signed Tcl Scripts, on page 2136](#)
- [How to Configure Signed Tcl Scripts, on page 2137](#)
- [Configuration Examples for Signed Tcl Script, on page 2150](#)
- [Additional References, on page 2154](#)
- [Feature Information for Signed Tcl Scripts, on page 2155](#)
- [Glossary, on page 2155](#)
- [Notices, on page 2156](#)

## Prerequisites for Signed Tcl Scripts

For this feature to work, the Cisco public key infrastructure (PKI) configuration trustpoint commands must be enabled.

## Restrictions for Signed Tcl Scripts

For this feature to work, you must be running the following:

- Cisco IOS Crypto image
- OpenSSL Version 0.9.7a or above
- Expect

# Information About Signed Tcl Scripts

The Signed Tcl Scripts feature introduces security for the Tcl scripts. This feature allows you to create a certificate to generate a digital signature and sign a Tcl script with that digital signature. This certificate examines the Tcl scripts prior to running them. The script is checked for a digital signature from Cisco. In addition, third parties may also sign a script with a digital signature. You may wish to sign your own internally developed Tcl scripts or you could use a script developed by a third party. If the script contains the correct digital signature, it is believed to be authentic and runs with full access to the Tcl interpreter. If the script does not contain the digital signature, the script may be run in a limited mode, known as Safe Tcl mode, or may not run at all.

To create and use signed Tcl scripts, you should understand the following concepts:

## Cisco PKI

Cisco PKI provides certificate management to support security protocols such as IP security (IPsec), secure shell (SSH), and secure socket layer (SSL). A PKI is composed of the following entities:

- Peers communicating on a secure network
- At least one certification authority (CA) that grants and maintains certificates
- Digital certificates, which contain information such as the certificate validity period, peer identity information, encryption keys that are used for secure communication, and the signature of the issuing CA
- An optional registration authority (RA) to offload the CA by processing enrollment requests
- A distribution mechanism (such as Lightweight Directory Access Protocol [LDAP] or HTTP) for certificate revocation lists (CRLs)

PKI provides you with a scalable, secure mechanism for distributing, managing, and revoking encryption and identity information in a secured data network. Every routing device participating in the secured communication is enrolled in the PKI in a process where the routing device generates a Rivest, Shamir, and Adelman (RSA) key pair (one private key and one public key) and has its identity validated by a trusted routing device (also known as a CA or trustpoint).

After each routing device enrolls in a PKI, every peer (also known as an end host) in a PKI is granted a digital certificate that has been issued by a CA. When peers must negotiate a secured communication session, they exchange digital certificates. Based on the information in the certificate, a peer can validate the identity of another peer and establish an encrypted session with the public keys contained in the certificate.

## RSA Key Pair

An RSA key pair consists of a public key and a private key. When setting up your PKI, you must include the public key in the certificate enrollment request. After the certificate has been granted, the public key is included in the certificate so that peers can use it to encrypt data that is sent to the device. The private key is kept on the device and used both to decrypt the data sent by peers and to digitally sign transactions when negotiating with peers.

RSA key pairs contain a key modulus value. The modulus determines the size of the RSA key. The larger the modulus, the more secure the RSA key. However, keys with large modulus values take longer to generate, and encryption and decryption operations take longer with larger keys.

## Certificate and Trustpoint

A certification authority (CA), also known as a trustpoint, manages certificate requests and issues certificates to participating network devices. These services (managing certificate requests and issuing certificates) provide centralized key management for the participating devices and are explicitly trusted by the receiver to validate identities and to create digital certificates. Before any PKI operations can begin, the CA generates its own public key pair and creates a self-signed CA certificate; thereafter, the CA can sign certificate requests and begin peer enrollment for the PKI.

You can use a CA provided by a third-party CA vendor, or you can use an internal CA, which is the Cisco Certificate Server.

## How to Configure Signed Tcl Scripts

### Generating a Key Pair

The key pair consists of a private key and a public key. The private key is intended to be kept private, accessible only to the creator. The public key is generated from the private key and is intended to be known to the public.

To generate a key pair, use the **openssl genrsa** command and then the **openssl rsa** command.

#### SUMMARY STEPS

1. **openssl genrsa -out** *private-key-file* *bit-length*
2. **ls -l**
3. **openssl rsa -in** *private-key-file* **-pubout -out** *public-key-file*
4. **ls -l**

#### DETAILED STEPS

**Step 1** **openssl genrsa -out** *private-key-file* *bit-length*

This command generates a private key that is *bit-length* bits long and writes the key to the *private-key-file* file.

```
Host% openssl genrsa -out privkey.pem 2048
```

#### Example:

```
Generating RSA private key, 2048 bit long modulus
.....+++
.....+++
e is 65537 (0x10001)
```

**Step 2** **ls -l**

This command displays detailed information about each file in the current directory, including the permissions, owners, size, and when last modified.

**Example:**

```
Host% ls -l
total 8
-rw-r--r-- 1 janedoe eng12 1679 Jun 12 14:55 privkey.pem
```

The `privkey.pem` file contains the private key generated using the `openssl genrsa` command.

**Step 3** `openssl rsa -in private-key-file -pubout -out public-key-file`

This command generates a public key based on the specified private key in the `private-key-file` file and writes the public key to the `public-key-file` file.

**Example:**

```
Host% openssl rsa -in privkey.pem -pubout -out pubkey.pem
writing RSA key
```

**Step 4** `ls -l`

This command displays detailed information about each file in the current directory, including the permissions, owners, size, and when last modified.

**Example:**

```
Host% ls -l
total 16
-rw-r--r-- 1 janedoe eng12 1679 Jun 12 14:55 privkey.pem
-rw-r--r-- 1 janedoe eng12 451 Jun 12 14:57 pubkey.pem
```

The `pubkey.pem` file contains the public key generated from the private key using the `openssl rsa` command.

## Generating a Certificate

Perform this task to generate a certificate. To generate an X.509 certificate, use the `openssl req` command.

### SUMMARY STEPS

1. `openssl req -new -x509 -key private-key-file -out certificate-file -days expiration-days`
2. `ls -l`

### DETAILED STEPS

**Step 1** `openssl req -new -x509 -key private-key-file -out certificate-file -days expiration-days`

This command creates an X.509 certificate, with full access to a private key that is stored in the `private-key-file` file, and stores the certificate in the `certificate-file` file. The certificate is configured to expire in `expiration-days` days.

To complete the command, enter the following Distinguished Name (DN) information when prompted:

- Country name
- State or province name
- Organization name
- Organizational unit name
- Common name
- Email address

At each prompt, text enclosed in square brackets indicates the default value that will be used if you do not enter a value before you press Enter.

This example shows how to create an X.509 certificate that has full access to the private key in the `privkey.pem` file. The certificate is written to the `cert.pem` file and will expire 1095 days after the creation date.

#### Example:

```
Host% openssl req -new -x509 -key privkey.pem -out cert.pem -days 1095
```

You are about to be asked to enter information that will be incorporated into your certificate request.

What you are about to enter is what is called a Distinguished Name or a DN.

There are quite a few fields but you can leave some blank

For some fields there will be a default value, If you enter '.', the field will be left blank.

-----

Country Name (2 letter code) [GB]:**US**

State or Province Name (full name) [Berkshire]:**California**

Locality Name (eg, city) [Newbury]:**San Jose**

Organization Name (eg, company) [My Company Ltd]:**Cisco Systems, Inc.**

Organizational Unit Name (eg, section) []:**DEPT\_ACCT**

Common Name (eg, your name or your server's hostname) []:**Jane**

Email Address []:**janedoe@company.com**

## Step 2 **ls -l**

This command displays detailed information about each file in the current directory, including the permissions, owners, size, and when last modified.

#### Example:

```
Host% ls -l
```

```
total 24
-rw-r--r-- 1 janedoe eng12 1659 Jun 12 15:01 cert.pem
-rw-r--r-- 1 janedoe eng12 1679 Jun 12 14:55 privkey.pem
-rw-r--r-- 1 janedoe eng12 451 Jun 12 14:57 pubkey.pem
```

The `cert.pem` file contains the X.509 certificate created using the `openssl req` command.

## Signing the Tcl Scripts

Perform this task to sign the Tcl scripts. You will need to sign the Tcl file and output in OpenSSL document in pkcs7 (PKCS#7) format.

To sign the Tcl file, use the **openssl smime** command with the **-sign** keyword.

### SUMMARY STEPS

1. **openssl smime -sign -in** *tcl-file* **-out** *signed-tcl-file* **-signer** *certificate-file* **-inkey** *private-key-file* **-outform DER -binary**
2. **ls -l**

### DETAILED STEPS

**Step 1** **openssl smime -sign -in** *tcl-file* **-out** *signed-tcl-file* **-signer** *certificate-file* **-inkey** *private-key-file* **-outform DER -binary**

This command signs the Tcl filename *tcl-file* using the certificate stored in *certificate-file* and the private key stored in *private-key-file* file and then writes the signed Tcl file in DER PKCS#7 format to the *signed-tcl-file* file.

**Example:**

```
Host% openssl smime -sign -in hello -out hello.pk7 -signer cert.pem -inkey privkey.pem -outform DER -binary
```

**Step 2** **ls -l**

This command displays detailed information about each file in the current directory, including the permissions, owners, size, and when last modified.

**Example:**

```
Host% ls -l

total 40
-rw-r--r-- 1 janedoe eng12 1659 Jun 12 15:01 cert.pem
-rw-r--r-- 1 janedoe eng12 115 Jun 13 10:16 hello
-rw-r--r-- 1 janedoe eng12 1876 Jun 13 10:16 hello.pk7
-rw-r--r-- 1 janedoe eng12 1679 Jun 12 14:55 privkey.pem
-rw-r--r-- 1 janedoe eng12 451 Jun 12 14:57 pubkey.pem
```

The *hello.pk7* file contains the signed Tcl file created by the **openssl smime** command from the unsigned Tcl file named *hello* and using the X.509 certificate in the *cert.pem* file.

## Verifying the Signature

Perform this task to verify that the signature matches the data, use the **openssl smime** command with the **-verify** keyword. The original Tcl content must be provided in the input file, because the file does not have the original content.

## SUMMARY STEPS

1. `openssl smime -verify -in signed-tcl-file -CAfile certificate-file -inform DER -content tcl-file`
2. `ls -l`

## DETAILED STEPS

### Step 1 `openssl smime -verify -in signed-tcl-file -CAfile certificate-file -inform DER -content tcl-file`

This command verifies the signed Tcl file stored in DER PKCS#7 format in *signed-tcl-file* using the trusted Certificate Authority (CA) certificates in *certificate-file* and then writes the detached content to the file *tcl-file*.

The following example shows how to verify the signature with the input file `hello.pk7`:

#### Example:

```
Host% openssl smime -verify -in hello.pk7 -CAfile cert.pem -inform DER -content hello

puts hello
puts "argc = $argc"
puts "argv = $argv"
puts "argv0 = $argv0"
puts "tcl_interactive = $tcl_interactive"
Verification successful
```

**Note** The SSL command page describes `-in filename` as the input message to be encrypted or signed or the MIME message to be decrypted or verified. For more information, go to <http://www.openssl.org/>.

### Step 2 `ls -l`

This command displays detailed information about each file in the current directory, including the permissions, owners, size, and when last modified.

#### Example:

```
Host% ls -l

total 40
-rw-r--r-- 1 janedoe eng12 1659 Jun 13 10:18 cert.pem
-rw-r--r-- 1 janedoe eng12 115 Jun 13 10:17 hello
-rw-r--r-- 1 janedoe eng12 1876 Jun 13 10:16 hello.pk7
-rw-r--r-- 1 janedoe eng12 1679 Jun 12 14:55 privkey.pem
-rw-r--r-- 1 janedoe eng12 451 Jun 12 14:57 pubkey.pem
```

The `hello` file contains the content detached from the signed Tcl file `hello.pk7` by running the `openssl smime` command with the `-verify` keyword. If the verification was successful, the signer's certificates are written to the X.509 certificate in the `cert.pem` file.

## Converting the Signature into Nonbinary Data

Perform this task to convert the signature from binary to nonbinary data.

## SUMMARY STEPS

1. `xxd -ps signed-tcl-file > nonbinary-signature-file`
2. Create a script that displays **#Cisco Tcl Signature V1.0** in the first line and inserts a comment character (#) at the beginning of each line of the input file and writes each line to a file whose name is formed by appending the text string “\_sig” to the name of the input file.
3. Run the script, supplying the name of the file containing the nonbinary signature file (*nonbinary-signature-file*) as the input argument.
4. `ls -l`
5. `cat signed-tcl-file commented-nonbinary-signature-file > signed-tcl-script`
6. `cat signed-tcl-script`

## DETAILED STEPS

**Step 1** `xxd -ps signed-tcl-file > nonbinary-signature-file`

This command converts the signature in *signed-tcl-file* from binary to nonbinary data and stores it as a hexadecimal dump in the file *nonbinary-signature-file*.

### Example:

```
Host% xxd -ps hello.pk7 > hello.hex
```

**Step 2** Create a script that displays **#Cisco Tcl Signature V1.0** in the first line and inserts a comment character (#) at the beginning of each line of the input file and writes each line to a file whose name is formed by appending the text string “\_sig” to the name of the input file.

In this example the `cat` command is used to display the contents of the script file named `my_append`.

### Example:

```
Host% cat my_append

#!/usr/bin/env expect
set my_first {#Cisco Tcl Signature V1.0}
set newline {}
set my_file [lindex $argv 0]
set my_new_file ${my_file}_sig
set my_new_handle [open $my_new_file w]
set my_handle [open $my_file r]
puts $my_new_handle $newline
puts $my_new_handle $my_first
foreach line [split [read $my_handle] "\n"] {
 set new_line (#)
 append new_line $line
 puts $my_new_handle $new_line
}

close $my_new_handle
close $my_handle
```

**Step 3** Run the script, supplying the name of the file containing the nonbinary signature file (*nonbinary-signature-file*) as the input argument.

In this example, the `my_append` script is run with the nonbinary signature file `hello.hex` specified as input. The output file will be named `hello.hex_sig`.



**Example:**

```
Host% my_append hello.hex
```

**Step 4** **ls -l**

This command displays detailed information about each file in the current directory, including the permissions, owners, size, and when last modified.

**Example:**

```
Host% ls -l
```

```
total 80
-rw-r--r-- 1 janedoe eng12 1659 Jun 13 10:18 cert.pem
-rw-r--r-- 1 janedoe eng12 115 Jun 13 10:17 hello
-rw-r--r-- 1 janedoe eng12 3815 Jun 13 10:20 hello.hex
-rw-r--r-- 1 janedoe eng12 3907 Jun 13 10:22 hello.hex_sig
-rw-r--r-- 1 janedoe eng12 1876 Jun 13 10:16 hello.pk7
-rwxr--r-- 1 janedoe eng12 444 Jun 13 10:22 my_append
-rw-r--r-- 1 janedoe eng12 1679 Jun 12 14:55 privkey.pem
-rw-r--r-- 1 janedoe eng12 451 Jun 12 14:57 pubkey.pem
```

The hello.hex file contains nonbinary data (stored as a hexadecimal dump) converted from the binary signature in the signed Tcl file hello.pk7. The my\_append file contains the script that inserts a comment character at the beginning of each line of the input file. The hello.hex\_sig file is the file created by running the my\_append script on the nonbinary signature file.

**Step 5** **cat signed-tcl-file commented-nonbinary-signature-file > signed-tcl-script**

This command appends the contents of the nonbinary signature file (*commented-nonbinary-signature-file*) to the signed Tcl file stored in DER PKCS#7 format (in the *signed-tcl-file* file). The concatenated output is written to the file *signed-tcl-script*.

**Example:**

```
Host% cat hello hello.hex_sig > hello.tcl
```

**Step 6** **cat signed-tcl-script**

This command displays the contents of the file *signed-tcl-script*, which is the concatenation of content detached from the signed Tcl file and the nonbinary signature file.

**Example:**

```
Host% cat hello.tcl
```

```
puts hello
puts "argc = $argc"
puts "argv = $argv"
puts "argv0 = $argv0"
puts "tcl_interactive = $tcl_interactive"
#Cisco Tcl Signature V1.0
#3082075006092a864886f70d010702a08207413082073d020101310b3009
#06052b0e03021a0500300b06092a864886f70d010701a08204a13082049d
#30820385a003020102020100300d06092a864886f70d0101040500308195
#310b3009060355040613025553311330110603550408130a43616c69666f
#726e69613111300f0603550407130853616e204a6f7365311c301a060355
#040a1313436973636f2053797374656d732c20496e632e310e300c060355
#040b13054e53535447310d300b060355040313044a6f686e3121301f0609
#2a864886f70d01090116126a6c6175746d616e40636973636f2e636f6d30
```

```
#1e170d3037303631323232303134335a170d313030363131323230313433
#5a308195310b3009060355040613025553311330110603550408130a4361
#6c69666f726e69613111300f0603550407130853616e204a6f7365311c30
#1a060355040a1313436973636f2053797374656d732c20496e632e310e30
#0c060355040b13054e53535447310d300b060355040313044a6f686e3121
#301f06092a864886f70d01090116126a6c6175746d616e40636973636f2e
#636f6d30820122300d06092a864886f70d010105000382010f00308201
#0a0282010100a751eb5ec1f3009738c88a55987c07b759c36f3386342283
#67ea20a89d9483ae85e0c63eeded8ab3eb7a08006689f09136f172183665
#c971099ba54e77ab47706069bbefaaab8c50184396350e4cc870c4c3f477
#88c55c52e2cf411f05b59f0eac0678ff5cc238fdce2263a9fc6b6c244b8
#ffaead865c19c3d3172674a13b24c8f2c01dd8b1bd491c13e84e29171b85
#f28155d81ac8c69bb25ca23c2921d85fbf745c106e7aff93c72316cbc654
#4a34ea88174a8ba7777fa60662974e1fbac85a0f0aeac925dba6e5e850b8
#7caffce2fe8bb04b61b62f532b5893c081522d538005df81670b931b0ad0
#e1e76ae648f598a9442d5d0976e67c8d55889299147d0203010001a381f5
#3081f2301d0603551d0e04160414bc34132be952ff8b9e1af3b93140a255
#e54a667c3081c20603551d230481ba3081b78014bc34132be952ff8b9e1a
#f3b93140a255e54a667ca1819ba48198308195310b300906035504061302
#5553311330110603550408130a43616c69666f726e69613111300f060355
#0407130853616e204a6f7365311c301a060355040a1313436973636f2053
#797374656d732c20496e632e310e300c060355040b13054e53535447310d
#300b060355040313044a6f686e3121301f06092a864886f70d0109011612
#6a6c6175746d616e40636973636f2e636f6d820100300c0603551d130405
#30030101ff300d06092a864886f70d010104050003820101000c83c1b074
#6720929c9514af6d5df96f0a95639f047c40a607c83d8362507c58fa7f84
#aa699ec5e5bef61b2308297a0662c653ff446acfb6f5cb2dd162d939338
#a5e4d78a5c45021e5d4dbabb8784efbf50cab0f5125d164487b31f5cf933
#a9f68f82cd111cbab1739d7f372ec460a7946882874b0a0f22dd53acb62
#a944a15e52e54a24341b3b8a820f23a5bc7ea7b2278bb56838b8a4051926
#af9c167274ff8449003a4e012bcf4f4b3e280f85209249a390d14df47435
#35efabc6720ea3d56803a84a2163db4478ae19d7d987ef6971c8312e280a
#aac0217d4fe620c6582a48faa8ea5e3726a99012e1d55f8d61b066381f77
#4158d144a43fb536c77d6a318202773082027302010130819b308195310b
#3009060355040613025553311330110603550408130a43616c69666f726e
#69613111300f0603550407130853616e204a6f7365311c301a060355040a
#1313436973636f2053797374656d732c20496e632e310e300c060355040b
#13054e53535447310d300b060355040313044a6f686e3121301f06092a86
#4886f70d01090116126a6c6175746d616e40636973636f2e636f6d020100
#300906052b0e03021a0500a081b1301806092a864886f70d010903310b06
#092a864886f70d010701301c06092a864886f70d010905310f170d303730
#3631333137313634385a302306092a864886f70d01090431160414372cb3
#72dc607990577fd0426104a42ee4158d2b305206092a864886f70d01090f
#31453043300a06082a864886f70d0307300e06082a864886f70d03020202
#0080300d06082a864886f70d0302020140300706052b0e030207300d0608
#2a864886f70d0302020128300d06092a864886f70d010101050004820100
#72db6898742f449b26d3ac18f43a1e7178834fb05ad13951bf042e127eea
#944b72b96f3b8ecf7eb52f3d0e383bf63651750223efe69eae04287c9dae
#b1f31209444108b31d34e46654c6c3cc10b5baba887825c224ec6f376d49
#00ff7ab2d9f88402dab9a2c2ab6aa3ecceef5a594bdc7d3a822c55e7daa
#aa0c2b067e06967f22a20e406fe21d9013ecc6bd9cd6d402c2749f8bea61
#9f8f87acfb9e10d6ce91502e34629adca6ee855419afafe6a8233333e14
#ad4c107901d1f2bca4d7ffaadddbc54192a25da662f8b8509782c76977b8
#94879453fbb00486ccc55f88db50fcc149bae066916b350089cde51a6483
#2ec14019611720fc5bbe2400f24225fc
```

## Configuring the Device with a Certificate

Perform this task to configure the device with a certificate.

### Before you begin

You must already have a Cisco IOS Crypto image; otherwise you cannot configure a certificate.

### SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **crypto pki trustpoint** *name*
4. **enrollment terminal**
5. **exit**
6. **crypto pki authenticate** *name*
7. At the prompt, enter the base-encoded CA certificate.
8. **scripting tcl secure-mode**
9. **scripting tcl trustpoint name** *name*
10. **scripting tcl trustpoint untrusted** {**execute** | **safe-execute** | **terminate**}
11. **exit**
12. **tclsafe**

### DETAILED STEPS

---

**Step 1**      **enable**

Enables privileged EXEC mode. Enter your password if prompted.

**Example:**

```
Device> enable
```

**Step 2**      **configure terminal**

Enters global configuration mode.

**Example:**

```
Device# configure terminal
```

**Step 3**      **crypto pki trustpoint** *name*

Declares the device is to use the Certificate Authority (CA) *mytrust* and enters ca-trustpoint configuration mode.

**Example:**

```
Device(config)# crypto pki trustpoint mytrust
```

**Step 4**      **enrollment terminal**

Specifies manual cut-and-paste certificate enrollment. When this command is enabled, the device displays the certificate request on the console terminal, allowing you to enter the issued certificate on the terminal.

**Example:**

```
Device(ca-trustpoint)# enrollment terminal
```

**Step 5**      **exit**

Exits ca-trustpoint configuration mode and returns to global configuration mode.

**Example:**

```
Device(ca-trustpoint)# exit
```

**Step 6**      **crypto pki authenticate name**

Retrieves the CA certificate and authenticates it. Check the certificate fingerprint if prompted.

**Note**      Because the CA signs its own certificate, you should manually authenticate the public key of the CA by contacting the CA administrator when you perform this command.

**Example:**

```
Device(config)# crypto pki authenticate mytrust
```

**Step 7**      At the prompt, enter the base-encoded CA certificate.**Example:**

```
Enter the base 64 encoded CA certificate.
End with a blank line or the word "quit" on a line by itself
MIIIEuDCCA6CgAwIBAgIBADANBgkqhkiG9w0BAQQFADCBnjELMAkGA1UEBhMCVVMx
EzARBgNVBAGTCkNhbg1mb3JuaWEwETAPBgNVBACTCFNhbiBkb3NlMRwwGgYDVQK
ExNDaXNjbyBTeXN0ZW1zLCBjbmuMQ4wDAYDVQQLEWVVOU1NURzEWMBQGA1UEAxMN
Sm9obiBMXYV0bWFubjEhMB8GCSqGSIb3DQEJARYSAmxhdXRtYW5AY21zY28uY29t
MB4XDTA2MTEwNzE3NTgwMVoXDTA5MTEwNzE3NTgwMVowgZ4xCzAJBgNVBAYTA1VT
MRMwEQYDVQKIEwepDYWxpZm9ybmlhMREwDwYDVQQHEWhTYW4gSm9zZTEcMBoGA1UE
ChMTQ21zY28uY21zZGVtcywgSW5jLjEOMAwGA1UECXMFTlNTVEcxFjAUBGNVBAMT
DUpvaG4gTGFlbG1hbm4xITAFBgkqhkiG9w0BCQEWEmpsYXV0bWFuQGNpc2NvLmNv
bTCCASiDQYJKoZIhvcNAQEBBQADggEPADCCAQoCggEBALxtqTMCirMb+CdyWLuH
oWAM8CEJDwGgGL7MWBhoi3TSMd/ww2XBB9biBtd1H6jHsjCiOwAR5OorakwFpyf7
mvRj2PqJALs+Vn93VBKIG6rZU14+wdOx686BVddIZvEJQPbRoIYTz fazWV70aLMV
bd7/B7vF1SG1YK9y1tX9p9nZyZ0x470AXetwOaGinvlG7VNuTXaASBLUjCRZsIlz
SBrXedBz6+BuoWmlFK45EYSlag5Rt9RGXXMBqzx91iyhrJ3zDDmkExa45yKJET
mAgDVMcpeteJtif47UDZJK30g4MbMyx/c8WGhmJ54qRL9BZEPmDxMQkNP1018MA1
Q8sCAwEAaAOB/jCB+zAdBgNVHQ4EFgQU9/ToDvbMR3JfJ4xEa4X47oNFq5kwgcsG
A1UdIwSBwzCBwIAU9/ToDvbMR3JfJ4xEa4X47oNFq5mhgaSkgaEwgZ4xCzAJBgNV
BAYTA1VTMRMwEQYDVQKIEwepDYWxpZm9ybmlhMREwDwYDVQQHEWhTYW4gSm9zZTEc
MBoGA1UECXMFTlNTVEcxFjAUBGNVBAMT DUpvaG4gTGFlbG1hbm4xITAFBgkqhkiG9w0BCQEWEmpsYXV0bWFuQGNp
c2NvLmNvbYIBADAMBGNVHRMEBTADAQH/MA0GCSqGSIb3DQEBAUAA4IBAQBtEs/4
MQeN9pT+XPCPg2ObQU8y2AadI+I34YK+fdHsFOh68hZhpzTN2VpNEvkFXpADhgr
7DkNGtwTCLa481v70iNFViQVL+inNrZwWMxoTnUNCK7Hc5kHkXt6cj0mvsefVUzx
Xl70mauhESRvlmYwRjXsSrEILerZYsuv5HbFdand+/rErmp2HVyfdntLnKdSzmXJ
5lwE/Et2QtYNGor00BlLesowfslR3LhHi4wn+5is7mALgNw/NuTiUrlzH18OeB4m
wcpBIJsLaJu6ZUJQ17IqdsWsa3fHd5qq0/k8P9z0YAYrf3+MFQr4ibvsYvHLO087
o2JslgW4qz34pqNh
Certificate has the following attributes:
 Fingerprint MD5: 1E327DBB 330936EB 2FB8EACB 4FD1133E
 Fingerprint SHA1: EE7FF9F4 05148842 B9D50FAC D76FDC9C E0703246
% Do you accept this certificate? [yes/no]: yes
Trustpoint CA certificate accepted.
% Certificate successfully imported
```

**Step 8**      **scripting tcl secure-mode**

Enables signature verification of the interactive Tcl scripts.

```
Device(config)# scripting tcl secure-mode
```

**Step 9**     **scripting tcl trustpoint name** *name*

Associates an existing configured trustpoint name with a certificate to verify Tcl scripts.

```
Device(config)# scripting tcl trustpoint name mytrust
```

**Step 10**    **scripting tcl trustpoint untrusted** {**execute** | **safe-execute** | **terminate**}

(Optional) Allows the interactive Tcl scripts to run regardless of the scripts failing in the signature check or in untrusted mode using one of the three keywords: **execute**, **safe-execute**, or **terminate**.

- **execute** --Executes Tcl scripts even if the signature verification fails. If the **execute** keyword is configured, signature verification is not at all performed.

**Note**     Use of this keyword is usually not recommended because the signature verification is not at all performed.

The **execute** keyword is provided for internal testing purposes and to provide flexibility. For example, in a situation where a certificate has expired but the other configurations are valid and you want to work with the existing configuration, then you can use the execute keyword to work around the expired certificate.

- **safe-execute** --Allows the script to run in safe mode. You can use the tclsafe command and also enter the interactive Tcl shell safe mode to explore the safe mode Tcl commands that are available. In order to get a better understanding of what is available in this limited safe mode, use the tclsafe Exec command to explore the options.
- **terminate** --Stops any script from running and reverts to default behavior. The default policy is to terminate. When the last trustpoint name is removed, the untrusted action is also removed. The untrusted action cannot be entered until at least one trustpoint name is configured for Tcl.

The following example shows how to execute the Tcl script in safe mode using the **safe-execute** keyword when the signature verification fails.

```
Device(config)# scripting tcl trustpoint untrusted safe-execute
```

**Step 11**    **exit**

Exits global configuration mode and returns to privileged EXEC mode.

```
Device(config)# exit
```

**Step 12**    **tclsafe**

(Optional) Enables the interactive Tcl shell untrusted safe mode. This allows you to manually run Tcl commands from the Cisco command line interface in untrusted safe mode.

```
Device# tclsafe
```

**Example:**

---

## Verifying the Trustpoint

To display the trustpoints that are configured in the device, use the **show crypto pki trustpoints** command.

### SUMMARY STEPS

1. **enable**
2. **show crypto pki trustpoints**

### DETAILED STEPS

---

#### Step 1 **enable**

This command enables privileged EXEC mode.

**Example:**

```
Device> enable
```

#### Step 2 **show crypto pki trustpoints**

This command displays the trustpoints that are configured in the device.

**Example:**

```
Device# show
crypto pki trustpoints

Trustpoint mytrust:
 Subject Name:
 ea=janedoe@cisco.com
 cn=Jane
 ou=DEPT_ACCT
 o=Cisco
 l=San Jose
 st=California
 c=US
 Serial Number: 00
 Certificate configured.
```

---

## Verifying the Signed Tcl Script

To verify that the Signed Tcl Script is properly running, use the **debug crypto pki transactions** command and the **tclsh** command.

### SUMMARY STEPS

1. **enable**

2. **debug crypto pki transactions**
3. **tclsh flash:signed-tcl-file**

## DETAILED STEPS

### Step 1 enable

This command enables privileged EXEC mode.

#### Example:

```
Device> enable
```

### Step 2 debug crypto pki transactions

This command display debugging messages for the trace of interaction (message type) between the CA and the device.

#### Example:

```
Device# debug crypto pki transactions
Crypto PKI Trans debugging is on
```

### Step 3 tclsh flash:signed-tcl-file

This command executes the Tcl script in Tcl shell.

**Note** The file should be a signed Tcl file.

#### Example:

```
Device# tclsh flash:hello.tcl

hello
argc = 0
argv =
argv0 = flash:hello.tcl
tcl_interactive = 0
device#
*Apr 21 04:46:18.563: CRYPTO_PKI: locked trustpoint mytrust, refcount is 1
*Apr 21 04:46:18.563: The PKCS #7 message has 0 verified signers.
*Apr 21 04:46:18.563: CRYPTO_PKI: Success on PKCS7 verify!
*Apr 21 04:46:18.563: CRYPTO_PKI: unlocked trustpoint mytrust, refcount is 0
```

## What to Do Next

- To get an overview of Crypto, refer to the “Part 5: Implementing and Managing a PKI” section of the *Security Configuration Guide*.

# Configuration Examples for Signed Tcl Script

## Generating a Key Pair Example

The following example shows how to generate the key pair--a private key and a public key:

### Generate a Private Key: Example

```
Host% openssl genrsa -out privkey.pem 2048
Generating RSA private key, 2048 bit long modulus
.....+++
.....+++
e is 65537 (0x10001)
Host% ls -l
total 8
-rw-r--r-- 1 janedoe eng12 1679 Jun 12 14:55 privkey.pem
Host%
```

### Generate a Public Key from the Private Key

```
Host% openssl rsa -in privkey.pem -pubout -out pubkey.pem
writing RSA key
Host% ls -l
total 16
-rw-r--r-- 1 janedoe eng12 1679 Jun 12 14:55 privkey.pem
-rw-r--r-- 1 janedoe eng12 451 Jun 12 14:57 pubkey.pem
```

## Generating a Certificate Example

The following example shows how to generate a certificate:

```
Host% openssl req -new -x509 -key privkey.pem -out cert.pem -days 1095
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value, If you enter '.', the field will be left
blank.

Country Name (2 letter code) [GB]:US
State or Province Name (full name) [Berkshire]:California
Locality Name (eg, city) [Newbury]:San Jose
Organization Name (eg, company) [My Company Ltd]:Cisco Systems, Inc.
Organizational Unit Name (eg, section) []:DEPT_ACCT
Common Name (eg, your name or your server's hostname) []:Jane
Email Address []:janedoe@company.com
Host% ls -l
total 24
-rw-r--r-- 1 janedoe eng12 1659 Jun 12 15:01 cert.pem
-rw-r--r-- 1 janedoe eng12 1679 Jun 12 14:55 privkey.pem
-rw-r--r-- 1 janedoe eng12 451 Jun 12 14:57 pubkey.pem
```



## Signing the Tcl Scripts Example

The following example shows how to sign the Tcl scripts:

```
Host% openssl smime -sign -in hello -out hello.pk7 -signer cert.pem -inkey privkey.pem
-outform DER -binary
Host% ls -l
total 40
-rw-r--r-- 1 janedoe eng12 1659 Jun 12 15:01 cert.pem
-rw-r--r-- 1 janedoe eng12 115 Jun 13 10:16 hello
-rw-r--r-- 1 janedoe eng12 1876 Jun 13 10:16 hello.pk7
-rw-r--r-- 1 janedoe eng12 1679 Jun 12 14:55 privkey.pem
-rw-r--r-- 1 janedoe eng12 451 Jun 12 14:57 pubkey.pem
```

## Verifying the Signature Example

The following example shows how to verify the signature:

```
Host% openssl smime -verify -in hello.pk7 -CAfile cert.pem -inform DER -content hello
puts hello
puts "argc = $argc"
puts "argv = $argv"
puts "argv0 = $argv0"
puts "tcl_interactive = $tcl_interactive"
Verification successful
```

## Converting the Signature with Nonbinary Data Example

The following example shows how to convert the Tcl signature with nonbinary data:

```
#Cisco Tcl Signature V1.0
Then append the signature file to the end of the file.
Host% xxd -ps hello.pk7 > hello.hex
Host% cat my_append
#!/usr/bin/env expect
set my_first {#Cisco Tcl Signature V1.0}
set newline {}
set my_file [lindex $argv 0]
set my_new_file ${my_file}_sig
set my_new_handle [open $my_new_file w]
set my_handle [open $my_file r]

puts $my_new_handle $newline
puts $my_new_handle $my_first
foreach line [split [read $my_handle] "\n"] {
 set new_line {#}
 append new_line $line
 puts $my_new_handle $new_line
}

close $my_new_handle
close $my_handle
Host% my_append hello.hex
Host% ls -l
total 80
-rw-r--r-- 1 janedoe eng12 1659 Jun 12 15:01 cert.pem
-rw-r--r-- 1 janedoe eng12 115 Jun 13 10:16 hello
-rw-r--r-- 1 janedoe eng12 3815 Jun 13 10:20 hello.hex
```

## Converting the Signature with Nonbinary Data Example

```

-rw-r--r-- 1 janedoe eng12 3907 Jun 13 10:22 hello.hex_sig
-rw-r--r-- 1 janedoe eng12 1876 Jun 13 10:16 hello.pk7
-rwxr--r-- 1 janedoe eng12 444 Jun 13 10:22 my_append
-rw-r--r-- 1 janedoe eng12 1679 Jun 12 14:55 privkey.pem
-rw-r--r-- 1 janedoe eng12 451 Jun 12 14:57 pubkey.pem
Host% cat hello hello.hex_sig > hello.tcl
Host% cat hello.tcl
puts hello
puts "argc = $argc"
puts "argv = $argv"
puts "argv0 = $argv0"
puts "tcl_interactive = $tcl_interactive"
#Cisco Tcl Signature V1.0
#3082075006092a864886f70d010702a08207413082073d020101310b3009
#06052b0e03021a0500300b06092a864886f70d010701a08204a13082049d
#30820385a003020102020100300d06092a864886f70d0101040500308195
#310b3009060355040613025553311330110603550408130a43616c69666f
#726e69613111300f0603550407130853616e204a6f7365311c301a060355
#040a1313436973636f2053797374656d732c20496e632e310e300c060355
#040b13054e53535447310d300b060355040313044a6f686e3121301f0609
#2a864886f70d01090116126a6c6175746d616e40636973636f2e636f6d30
#1e170d3037303631323232303134335a170d313030363131323230313433
#5a308195310b3009060355040613025553311330110603550408130a4361
#6c69666f726e69613111300f0603550407130853616e204a6f7365311c30
#1a060355040a1313436973636f2053797374656d732c20496e632e310e30
#0c060355040b13054e53535447310d300b060355040313044a6f686e3121
#301f06092a864886f70d01090116126a6c6175746d616e40636973636f2e
#636f6d30820122300d06092a864886f70d010105000382010f00308201
#0a0282010100a751eb5ec1f3009738c88a55987c07b759c36f3386342283
#67ea20a89d9483ae85e0c63eeded8ab3eb7a08006689f09136f172183665
#c971099ba54e77ab47706069bbefaaab8c50184396350e4cc870c4c3f477
#88c55c52e2cf411f05b59f0eaec0678ff5cc238fdce2263a9fc6b6c244b8
#fffaead865c19c3d3172674a13b24c8f2c01dd8b1bd491c13e84e29171b85
#f28155d81ac8c69bb25ca23c2921d85fbf745c106e7aff93c72316cbc654
#4a34ea88174a8ba7777fa60662974e1fbac85a0f0aeac925dba6e5e850b8
#7caffce2fe8bb04b61b62f532b5893c081522d538005df81670b931b0ad0
#e1e76ae648f598a9442d5d0976e67c8d55889299147d0203010001a381f5
#3081f2301d0603551d0e04160414bc34132be952ff8b9e1af3b93140a255
#e54a667c3081c20603551d230481ba3081b78014bc34132be952ff8b9e1a
#f3b93140a255e54a667ca1819ba48198308195310b300906035504061302
#5553311330110603550408130a43616c69666f726e69613111300f060355
#0407130853616e204a6f7365311c301a060355040a1313436973636f2053
#797374656d732c20496e632e310e300c060355040b13054e53535447310d
#300b060355040313044a6f686e3121301f06092a864886f70d0109011612
#6a6c6175746d616e40636973636f2e636f6d820100300c0603551d130405
#30030101ff300d06092a864886f70d010104050003820101000c83c1b074
#6720929c9514af6d5df96f0a95639f047c40a607c83d8362507c58fa7f84
#aa699ec5e5bef61b2308297a0662c653ff446acfb6f5cb2dd162d939338
#a5e4d78a5c45021e5d4dbabb8784efbf50cab0f5125d164487b31f5cf933
#a9f68f82cd111cbab1739d7f372ec460a7946882874b0a0f22dd53acbd62
#a944a15e52e54a24341b3b8a820f23a5bc7ea7b2278bb56838b8a4051926
#af9c167274ff8449003a4e012bcf4f4b3e280f85209249a390d14df47435
#35efabce720ea3d56803a84a2163db4478ae19d7d987ef6971c8312e280a
#aac0217d4fe620c6582a48faa8ea5e3726a99012e1d55f8d61b066381f77
#4158d144a43fb536c77d6a318202773082027302010130819b308195310b
#3009060355040613025553311330110603550408130a43616c69666f726e
#69613111300f0603550407130853616e204a6f7365311c301a060355040a
#1313436973636f2053797374656d732c20496e632e310e300c060355040b
#13054e53535447310d300b060355040313044a6f686e3121301f06092a86
#4886f70d01090116126a6c6175746d616e40636973636f2e636f6d020100
#300906052b0e03021a0500a081b1301806092a864886f70d010903310b06
#092a864886f70d010701301c06092a864886f70d010905310f170d303730
#3631333137313634385a302306092a864886f70d01090431160414372cb3
#72dc607990577fd0426104a42ee4158d2b305206092a864886f70d01090f

```

```
#31453043300a06082a864886f70d0307300e06082a864886f70d03020202
#0080300d06082a864886f70d0302020140300706052b0e030207300d0608
#2a864886f70d0302020128300d06092a864886f70d010101050004820100
#72db6898742f449b26d3ac18f43a1e7178834fb05ad13951bf042e127eea
#944b72b96f3b8ecf7eb52f3d0e383bf63651750223efe69eae04287c9dae
#b1f31209444108b31d34e46654c6c3cc10b5baba887825c224ec6f376d49
#00ff7ab2d9f88402dab9a2c2ab6aa3ecceef5a594bdc7d3a822c55e7daa
#aa0c2b067e06967f22a20e406fe21d9013ecc6bd9cd6d402c2749f8bea61
#9f8f87acfb9e10d6ce91502e34629adca6ee855419afafe6a823333e14
#ad4c107901d1f2bca4d7ffaaddbc54192a25da662f8b8509782c76977b8
#94879453fbb00486ccc55f88db50fcc149bae066916b350089cde51a6483
#2ec14019611720fc5bbe2400f24225fc
```

## Configuring the Device with a Certificate Example

The following example shows how to configure the device with a certificate:

```
crypto pki trustpoint mytrust
 enrollment terminal
!
!
crypto pki authentication mytrust
crypto pki certificate chain mytrust
certificate ca 00
 308204B8 308203A0 A0030201 02020100 300D0609 2A864886 F70D0101 04050030
 819E310B 30090603 55040613 02555331 13301106 03550408 130A4361 6C69666F
 726E6961 3111300F 06035504 07130853 616E204A 6F736531 1C301A06 0355040A
 13134369 73636F20 53797374 656D732C 20496E63 2E310E30 0C060355 040B1305
 4E535354 47311630 14060355 0403130D 4A6F686E 204C6175 746D616E 6E312130
 1F06092A 864886F7 0D010901 16126A6C 6175746D 616E4063 6973636F 2E636F6D
 301E170D 30363131 31373137 35383031 5A170D30 39313131 36313735 3830315A
 30819E31 0B300906 03550406 13025553 31133011 06035504 08130A43 616C6966
 6F726E69 61311130 0F060355 04071308 53616E20 4A6F7365 311C301A 06035504
 0A131343 6973636F 20537973 74656D73 2C20496E 632E310E 300C0603 55040B13
 054E5353 54473116 30140603 55040313 0D4A6F68 6E204C61 75746D61 6E6E3121
 301F0609 2A864886 F70D0109 0116126A 6C617574 6D616E40 63697363 6F2E636F
 6D308201 22300D06 092A8648 86F70D01 01010500 0382010F 00308201 0A028201
 0100BC6D A933028A B31BF827 7258BB87 A1600CF0 21090F04 2080BECC 5818688B
 74D231DF F0C365C1 07D6E206 D7651FA8 C7B230A2 3B0011E4 EA2B6A4C 1F3F27FB
 9AF449D8 FA8900BB 3E567F77 5412881B AAD9525E 3EC1D3B1 EBCE8155 D74866F1
 0940F6D1 3A2613CD F6B3595E F468B315 6DDEF077 BBC5D521 B560AF72 D6D5FDA7
 D9D9C99D 31E3B380 5DEB7039 A1A29EF9 46ED536E 4D768048 12D48C24 59B08973
 481AD75D E741CD9E BE06EA16 9B514AE3 91184A56 A0E51B7D 4465D730 1AB3C7DD
 62CA1AC9 DF30C39A 41316B8E 72289113 98080354 C7297AD7 89B627F8 ED40D924
 ADF48383 1B332C7F 73C58686 6279E2A4 4BF41644 3E60F131 090D3F5D 25F0C025
 43CB0203 010001A3 81FE3081 FB301D06 03551D0E 04160414 F7F4E80E F6CC4772
 5F278C44 6B85F8EE 8345AB99 3081CB06 03551D23 0481C330 81C08014 F7F4E80E
 F6CC4772 5F278C44 6B85F8EE 8345AB99 A181A4A4 81A13081 9E310B30 09060355
 04061302 55533113 30110603 55040813 0A43616C 69666F72 6E696131 11300F06
 03550407 13085361 6E204A6F 7365311C 301A0603 55040A13 13436973 636F2053
 79737465 6D732C20 496E632E 310E300C 06035504 0B13054E 53535447 31163014
 06035504 03130D4A 6F686E20 4C617574 6D616E6E 3121301F 06092A86 4886F70D
 01090116 126A6C61 75746D61 6E406369 73636F2E 636F6D82 0100300C 0603551D
 13040530 030101FF 300D0609 2A864886 F70D0101 04050003 82010100 6D12CFF8
 31078DF6 94FE5CF0 8F83639B 414F32D8 069D23E2 37E182BE 7C31EC14 E87AF216
 61A6CCD3 37656934 4BE4157A 400E182B EC390D1A DC130A56 B8F35BFB D2234556
 24152FE8 A736B670 58CC684E 750D08AE C7739907 917B7A72 3D26BEC7 9F554CF1
 5E5EF499 ABA11124 55966616 AC9C52B2 B1082DEA D962CBAF E476C575 A9DDFBFA
 C4AE63F6 1D5C9F76 7B4B9CA7 52CE65C9 E65C04FC 4B7642D6 0D1A8AF4 38194B7A
 CA307EC9 51DCB847 8B8C27FB 98ACEE60 0B80DC3F 36E4E252 BD731F5F 0E781E26
 C1CA4120 9B0B689B BA654250 97B22A76 CC126B77 C7779AAA D3F93C3F DCF46006
 2B7F7F8C 150AF889 BBEC62F1 E53B4F3B A3626CD6 05B8AB3D F8A6A361
```

```

quit
archive
 log config
 scripting tcl trustpoint name mytrust
 scripting tcl secure-mode
!
!
end

```

## Additional References

The following sections provide references related to writing EEM policies Using the Cisco IOS CLI.

### Related Documents

| Related Topic                                                                                                  | Document Title                                                     |
|----------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------|
| EEM commands: complete command syntax, defaults, command mode, command history, usage guidelines, and examples | <a href="#">Cisco IOS Embedded Event Manager Command Reference</a> |
| Embedded Event Manager overview                                                                                | Embedded Event Manager Overview module                             |
| Embedded Event Manager policy writing using Tcl                                                                | Writing Embedded Event Manager Policies Using Tcl module           |
| Configuring enhanced object tracking                                                                           | Configuring Enhanced Object Tracking module                        |

### Standards

| Standard                                                                                              | Title |
|-------------------------------------------------------------------------------------------------------|-------|
| No new or modified standards are supported, and support for existing standards has not been modified. | --    |

### MIBs

| MIB                          | MIBs Link                                                                                                                                                                                                              |
|------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| CISCO-EMBEDDED-EVENT-MGR-MIB | To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL:<br><a href="http://www.cisco.com/go/mibs">http://www.cisco.com/go/mibs</a> |

### RFCs

| RFC                                                                                         | Title |
|---------------------------------------------------------------------------------------------|-------|
| No new or modified RFCs are supported, and support for existing RFCs has not been modified. | --    |

**Technical Assistance**

| Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             | Link                                                                                                                     |
|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------|
| <p>The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies.</p> <p>To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds.</p> <p>Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.</p> | <p><a href="http://www.cisco.com/cisco/web/support/index.html">http://www.cisco.com/cisco/web/support/index.html</a></p> |

## Feature Information for Signed Tcl Scripts

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to [www.cisco.com/go/cfn](http://www.cisco.com/go/cfn). An account on Cisco.com is not required.

*Table 218: Feature Information for Signed Tcl Scripts*

| Feature Name       | Releases  | Feature Information                                           |
|--------------------|-----------|---------------------------------------------------------------|
| Signed Tcl Scripts | 15.2(5)E1 | This feature was introduced and is supported only on c2960cx. |

## Glossary

CA--certification authority. Service responsible for managing certificate requests and issuing certificates to participating IPsec network devices. This service provides centralized key management for the participating devices and is explicitly trusted by the receiver to validate identities and to create digital certificates.

certificates--Electronic documents that bind a user's or device's name to its public key. Certificates are commonly used to validate a digital signature.

CRL--certificate revocation list. Electronic document that contains a list of revoked certificates. The CRL is created and digitally signed by the CA that originally issued the certificates. The CRL contains dates for when the certificate was issued and when it expires. A new CRL is issued when the current CRL expires.

IPsec--IP security

peer certificate--Certificate presented by a peer, which contains the peer's public key and is signed by the trustpoint CA.

PKI--public key infrastructure. System that manages encryption keys and identity information for components of a network that participate in secured communications.

RA--registration authority. Server that acts as a proxy for the CA so that CA functions can continue when the CA is offline. Although the RA is often part of the CA server, the RA could also be an additional application, requiring an additional device to run it.

RSA keys--Public key cryptographic system developed by Ron Rivest, Adi Shamir, and Leonard Adleman. An RSA key pair (a public and a private key) is required before you can obtain a certificate for your device.

SHA1--Secure Hash Algorithm 1

SSH--secure shell

SSL--secure socket layer

## Notices

The following notices pertain to this software license.

### OpenSSL Open SSL Project

This product includes software developed by the OpenSSL Project for use in the OpenSSL Toolkit ( <http://www.openssl.org/> ).

This product includes cryptographic software written by Eric Young (eay@cryptsoft.com).

This product includes software written by Tim Hudson (tjh@cryptsoft.com).

### License Issues

The OpenSSL toolkit stays under a dual license, i.e. both the conditions of the OpenSSL License and the original SSLeay license apply to the toolkit. See below for the actual license texts. Actually both licenses are BSD-style Open Source licenses. In case of any license issues related to OpenSSL please contact [openssl-core@openssl.org](mailto:openssl-core@openssl.org).

#### OpenSSL License:

Copyright © 1998-2007 The OpenSSL Project. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions, and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. All advertising materials mentioning features or use of this software must display the following acknowledgment: "This product includes software developed by the OpenSSL Project for use in the OpenSSL Toolkit ( <http://www.openssl.org/> )".

4. The names “OpenSSL Toolkit” and “OpenSSL Project” must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact [openssl-core@openssl.org](mailto:openssl-core@openssl.org).
5. Products derived from this software may not be called “OpenSSL” nor may “OpenSSL” appear in their names without prior written permission of the OpenSSL Project.
6. Redistributions of any form whatsoever must retain the following acknowledgment:

“This product includes software developed by the OpenSSL Project for use in the OpenSSL Toolkit (<http://www.openssl.org/>)”.

THIS SOFTWARE IS PROVIDED BY THE OpenSSL PROJECT “AS IS” AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE OpenSSL PROJECT OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

This product includes cryptographic software written by Eric Young ([eay@cryptsoft.com](mailto:eay@cryptsoft.com)). This product includes software written by Tim Hudson ([tjh@cryptsoft.com](mailto:tjh@cryptsoft.com)).

#### **Original SSLeay License:**

Copyright © 1995-1998 Eric Young ([eay@cryptsoft.com](mailto:eay@cryptsoft.com)). All rights reserved.

This package is an SSL implementation written by Eric Young ([eay@cryptsoft.com](mailto:eay@cryptsoft.com)).

The implementation was written so as to conform with Netscapes SSL.

This library is free for commercial and non-commercial use as long as the following conditions are adhered to. The following conditions apply to all code found in this distribution, be it the RC4, RSA, lhash, DES, etc., code; not just the SSL code. The SSL documentation included with this distribution is covered by the same copyright terms except that the holder is Tim Hudson ([tjh@cryptsoft.com](mailto:tjh@cryptsoft.com)).

Copyright remains Eric Young’s, and as such any Copyright notices in the code are not to be removed. If this package is used in a product, Eric Young should be given attribution as the author of the parts of the library used. This can be in the form of a textual message at program startup or in documentation (online or textual) provided with the package.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. All advertising materials mentioning features or use of this software must display the following acknowledgement:

“This product includes cryptographic software written by Eric Young ([eay@cryptsoft.com](mailto:eay@cryptsoft.com))”.

The word ‘cryptographic’ can be left out if the routines from the library being used are not cryptography-related.

1. If you include any Windows specific code (or a derivative thereof) from the apps directory (application code) you must include an acknowledgement: “This product includes software written by Tim Hudson (tjh@cryptsoft.com)”.

THIS SOFTWARE IS PROVIDED BY ERIC YOUNG “AS IS” AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The license and distribution terms for any publicly available version or derivative of this code cannot be changed. i.e. this code cannot simply be copied and put under another distribution license [including the GNU Public License].





## CHAPTER 103

# EEM CLI Library Command Extensions

All command-line interface (CLI) library command extensions belong to the `::cisco::eem` namespace.

This library provides users the ability to run CLI commands and get the output of the commands in Tcl. Users can use commands in this library to spawn an exec and open a virtual terminal channel to it, write the command to execute to the channel so that the command will be executed by exec, and read back the output of the command.

There are two types of CLI commands: interactive commands and non-interactive commands.

For interactive commands, after the command is entered, there will be a "Q&A" phase in which the device will ask for different user options, and the user is supposed to enter the answer for each question. Only after all the questions have been answered properly will the command run according to the user's options until completion.

For noninteractive commands, once the command is entered, the command will run to completion. To run different types of commands using an EEM script, different CLI library command sequences should be used, which are documented in the "Using the CLI Library to Run a Noninteractive Command" section and in the "Using the CLI Library to Run an Interactive Command" section in the `cli_write` Tcl command.

The vty lines are allocated from the pool of vty lines that are configured using the `line vty` CLI configuration command. EEM will use a vty line when a vty line is not being used by EEM and there are available vty lines. EEM will also use a vty line when EEM is already using a vty line and there are three or more vty lines available. Be aware that the connection will fail when fewer than three vty lines are available, preserving the remaining vty lines for Telnet use.

Your release may support XML-PI. For details about the XML-PI support, the new CLI library command extensions, and some examples of how to implement XML-PI, see EEM CLI Library XML-PI Support.

- [cli\\_close](#), on page 2160
- [cli\\_exec](#), on page 2160
- [cli\\_get\\_ttyname](#), on page 2161
- [cli\\_open](#), on page 2161
- [cli\\_read](#), on page 2162
- [cli\\_read\\_drain](#), on page 2162
- [cli\\_read\\_line](#), on page 2163
- [cli\\_read\\_pattern](#), on page 2163
- [cli\\_run](#), on page 2164
- [cli\\_run\\_interactive](#), on page 2165
- [cli\\_write](#), on page 2166

- [EEM 4.0 CLI Library XML-PI Support, on page 2169](#)
- [EEM CLI Library XML-PI Support, on page 2169](#)

## cli\_close

Closes the exec process and releases the vty and the specified channel handler connected to the command-line interface (CLI).

### Syntax

```
cli_close fd tty_id
```

### Arguments

|        |                                                                             |
|--------|-----------------------------------------------------------------------------|
| fd     | (Mandatory) The CLI channel handler.                                        |
| tty_id | (Mandatory) The TTY ID returned from the <b>cli_open</b> command extension. |

### Result String

None

### Set\_cerrno

Cannot close the channel.

## cli\_exec

Writes the command to the specified channel handler to execute the command. Then reads the output of the command from the channel and returns the output.

### Syntax

```
cli_exec fd cmd
```

### Arguments

|     |                                                               |
|-----|---------------------------------------------------------------|
| fd  | (Mandatory) The command-line interface (CLI) channel handler. |
| cmd | (Mandatory) The CLI command to execute.                       |

### Result String

The output of the CLI command executed.

### Set\_cerrno

Error reading the channel.

## cli\_get\_ttyname

Returns the real and pseudo TTY names for a given TTY ID.

### Syntax

```
cli_get_ttyname tty_id
```

### Arguments

|        |                                                                             |
|--------|-----------------------------------------------------------------------------|
| tty_id | (Mandatory) The TTY ID returned from the <b>cli_open</b> command extension. |
|--------|-----------------------------------------------------------------------------|

### Result String

```
pty %s tty %s
```

### Set\_cerrno

None

## cli\_open

Allocates a vty, creates an EXEC command-line interface (CLI) session, and connects the vty to a channel handler. Returns an array including the channel handler.



---

**Note** Each call to **cli\_open** initiates a Cisco IOS EXEC session that allocates a Cisco IOS vty line. The vty remains in use until the **cli\_close** routine is called. The vty lines are allocated from the pool of vty lines that are configured using the **line vty** CLI configuration command. EEM will use a vty line when a vty line is not being used by EEM and there are available vty lines. EEM will also use a vty line when EEM is already using a vty line and there are three or more vty lines available. Be aware that the connection will fail when fewer than three vty lines are available, preserving the remaining vty lines for Telnet use.

---

### Syntax

```
cli_open
```

### Arguments

None

### Result String

```
"tty_id {%s} pty {%d} tty {%d} fd {%d}"
```

| Event Type    | Description          |
|---------------|----------------------|
| <b>tty_id</b> | TTY ID.              |
| <b>pty</b>    | PTY device name.     |
| <b>tty</b>    | TTY device name.     |
| <b>fd</b>     | CLI channel handler. |

**Set\_cerrno**

- Cannot get pty for EXEC.
- Cannot create an EXEC CLI session.
- Error reading the first prompt.

## cli\_read

Reads the command output from the specified command-line interface (CLI) channel handler until the pattern of the device prompt occurs in the contents read. Returns all the contents read up to the match.

**Syntax**

```
cli_read fd
```

**Arguments**

|           |                                      |
|-----------|--------------------------------------|
| <b>fd</b> | (Mandatory) The CLI channel handler. |
|-----------|--------------------------------------|

**Result String**

All the contents read.

**Set\_cerrno**

Cannot get device name.




---

**Note** This Tcl command extension will block waiting for the device prompt to show up in the contents read.

---

## cli\_read\_drain

Reads and drains the command output of the specified command-line interface (CLI) channel handler. Returns all the contents read.

**Syntax**

```
cli_read_drain fd
```

**Arguments**

|    |                                      |
|----|--------------------------------------|
| fd | (Mandatory) The CLI channel handler. |
|----|--------------------------------------|

**Result String**

All the contents read.

**Set\_cerrno**

None

## cli\_read\_line

Reads one line of the command output from the specified command-line interface (CLI) channel handler. Returns the line read.

**Syntax**

```
cli_read_line fd
```

**Arguments**

|    |                                      |
|----|--------------------------------------|
| fd | (Mandatory) The CLI channel handler. |
|----|--------------------------------------|

**Result String**

The line read.

**Set\_cerrno**

None



---

**Note** This Tcl command extension will block waiting for the end of line to show up in the contents read.

---

## cli\_read\_pattern

Reads the command output from the specified command-line interface (CLI) channel handler until the pattern that is to be matched occurs in the contents read. Returns all the contents read up to the match.




---

**Note** The pattern matching logic attempts a match by looking at the command output data as it is delivered from the Cisco IOS command. The match is always done on the most recent 256 characters in the output buffer unless there are fewer characters available, in which case the match is done on fewer characters. If more than 256 characters in the output buffer are required for the match to succeed, the pattern will not match.

---

**Syntax**

```
cli_read_pattern fd ptn
```

**Arguments**

|     |                                                                                         |
|-----|-----------------------------------------------------------------------------------------|
| fd  | (Mandatory) The CLI channel handler.                                                    |
| ptn | (Mandatory) The pattern to be matched when reading the command output from the channel. |

**Result String**

All the contents read.

**Set\_cerrno**

None




---

**Note** This Tcl command extension will block waiting for the specified pattern to show up in the contents read.

---

## cli\_run

Iterates over the items in the clist and assumes that each one is a command-line-interface (CLI) command to be executed in the enable mode. On success, returns the output of all executed commands and on failure, returns error from the failure.

**Syntax**

```
cli_run clist
```

**Arguments**

|       |                                                  |
|-------|--------------------------------------------------|
| clist | (Mandatory) The list of commands to be executed. |
|-------|--------------------------------------------------|

**Result String**

Output of all the commands that are executed or an error message.

**Set \_cerrno**

None.

**Sample Usage**

The following example shows how to use the **cli\_run** command extension.

```
set clist [list {sh run} {sh ver} {sh event man pol reg}]
cli_run { clist }
```

# cli\_run\_interactive

Provides a sublist to the clist which has three items. On success, returns the output of all executed commands and on failure, returns error from the failure. Also uses arrays when possible as a way of making things easier to read later by keeping expect and reply separated.

**Syntax**

```
cli_run_interactive clist
```

**Arguments**

|       |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |
|-------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| clist | <p>(Mandatory) List of three items:</p> <ul style="list-style-type: none"> <li>• command– Command to be executed</li> <li>• expect– A regular expression pattern match for the expected reply prompt</li> <li>• responses– A list of possible responses to the reply prompt constructed as an array of two items: <ul style="list-style-type: none"> <li>• expect– A regular expression pattern match for a possible reply prompt</li> <li>• reply- A reply for that expected prompt</li> </ul> </li> </ul> |
|-------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|

**Result String**

Output of all the commands that are executed or an error message. As each command is executed its output is appended to a result variable. Upon exhaustion of the input list, the CLI channel is closed and the aggregate result is returned.

**Set \_cerrno**

None.

**Sample Usage**

The following example shows how to clear counters for interface fa0/0 use the cli\_run\_interactive command extension.

```

set cmdarr(command) "clear counters fa0/0"
set cmdarr(responses) [list]
set resps(expect) {[confirm]}
set resps(reply) "y"
lappend cmdarr(responses) [array get resps]
set rc [catch {cli_run_interactive [list [array get cmdarr]]} result]

```

Possible errors raised include:

- cannot get pty for exec
- cannot spawn exec
- error reading the first prompt
- error reading the channel
- cannot close channel

## cli\_write

Writes the command that is to be executed to the specified CLI channel handler. The CLI channel handler executes the command.

### Syntax

```
cli_write fd cmd
```

### Arguments

|     |                                         |
|-----|-----------------------------------------|
| fd  | (Mandatory) The CLI channel handler.    |
| cmd | (Mandatory) The CLI command to execute. |

### Result String

None

### Set\_cerrno

None

### Sample Usage

As an example, use configuration CLI commands to bring up Ethernet interface 1/0:

```

if [catch {cli_open} result] {
puts stderr $result
exit 1
} else {
array set cli1 $result
}
if [catch {cli_exec $cli1(fd) "en"} result] {
puts stderr $result
exit 1
}

```



```

}
if [catch {cli_exec $cli1(fd) "config t"} result] {
puts stderr $result
exit 1
}
if [catch {cli_exec $cli1(fd) "interface Ethernet1/0"} result] {
puts stderr $result
exit 1
}
if [catch {cli_exec $cli1(fd) "no shut"} result] {
puts stderr $result
exit 1
}
if [catch {cli_exec $cli1(fd) "end"} result] {
puts stderr $result
exit 1
}
if [catch {cli_close $cli1(fd) $cli1(tty_id)} } result] {
puts stderr $result
exit 1
}

```

### Using the CLI Library to Run a Noninteractive Command

To run a noninteractive command, use the **cli\_exec** command extension to issue the command, and then wait for the complete output and the device prompt. For example, the following shows the use of configuration CLI commands to bring up Ethernet interface 1/0:

```

if [catch {cli_open} result] {
error $result $errorInfo
} else {
set fd $result
}
if [catch {cli_exec $fd "en"} result] {
error $result $errorInfo
}
if [catch {cli_exec $fd "config t"} result] {
error $result $errorInfo
}
if [catch {cli_exec $fd "interface Ethernet1/0"} result] {
error $result $errorInfo
}
if [catch {cli_exec $fd "no shut"} result] {
error $result $errorInfo
}
if [catch {cli_exec $fd "end"} result] {
error $result $errorInfo
}
if [catch {cli_close $fd} result] {
error $result $errorInfo
}
}

```

### Using the CLI Library to Run an Interactive Command

To run interactive commands, three phases are needed:

- Phase 1: Issue the command using the **cli\_write** command extension.
- Phase 2: Q&A Phase. Use the **cli\_read\_pattern** command extension to read the question (the regular pattern that is specified to match the question text) and the **cli\_write** command extension to write back the answers alternately.

- Phase 3: Noninteractive phase. All questions have been answered, and the command will run to completion. Use the **cli\_read** command extension to wait for the complete output of the command and the device prompt.

For example, use CLI commands to do squeeze bootflash: and save the output of this command in the Tcl variable `cmd_output`.

```

if [catch {cli_open} result] {
error $result $errorInfo
} else {
array set cli1 $result
}
if [catch {cli_exec $cli1(fd) "en"} result] {
error $result $errorInfo
}

Phase 1: issue the command
if [catch {cli_write $cli1(fd) "squeeze bootflash:"} result] {
error $result $errorInfo
}

Phase 2: Q&A phase
wait for prompted question:
All deleted files will be removed. Continue? [confirm]
if [catch {cli_read_pattern $cli1(fd) "All deleted"} result] {
error $result $errorInfo
}
write a newline character
if [catch {cli_write $cli1(fd) "\n"} result] {
error $result $errorInfo
}
wait for prompted question:
Squeeze operation may take a while. Continue? [confirm]
if [catch {cli_read_pattern $cli1(fd) "Squeeze operation"} result] {
error $result $errorInfo
}
write a newline character
if [catch {cli_write $cli1(fd) "\n"} result] {
error $result $errorInfo
}

Phase 3: noninteractive phase
wait for command to complete and the router prompt
if [catch {cli_read $cli1(fd) } result] {
error $result $errorInfo
} else {
set cmd_output $result
}
if [catch {cli_close $cli1(fd) $cli1(tty_id)} result] {
error $result $errorInfo
}

```

The following example causes a device to be reloaded using the CLI **reload** command. Note that the EEM **action\_reload** command accomplishes the same result in a more efficient manner, but this example is presented to illustrate the flexibility of the CLI library for interactive command execution.

```

1. execute the reload command
if [catch {cli_open} result] {
error $result $errorInfo
} else {
array set cli1 $result

```

```

}
if [catch {cli_exec $cli1(fd) "en"} result] {
 error $result $errorInfo
}
if [catch {cli_write $cli1(fd) "reload"} result] {
 error $result $errorInfo
} else {
 set cmd_output $result
}
if [catch {cli_read_pattern $cli1(fd) ".*(System configuration has been modified. Save\\? \\[yes/no\\]:)"} result] {
 error $result $errorInfo
} else {
 set cmd_output $result
}
if [catch {cli_write $cli1(fd) "no"} result] {
 error $result $errorInfo
} else {
 set cmd_output $result
}
if [catch {cli_read_pattern $cli1(fd) ".*(Proceed with reload\\? \\[confirm\\])"} result]
{
 error $result $errorInfo
} else {
 set cmd_output $result
}
if [catch {cli_write $cli1(fd) "y"} result] {
 error $result $errorInfo
} else {
 set cmd_output $result
}
if [catch {cli_close $cli1(fd) $cli1(tty_id)} result] {
 error $result $errorInfo
}
}

```

## EEM 4.0 CLI Library XML-PI Support

### EEM CLI Library XML-PI Support

XML Programmatic Interface (XML-PI) was introduced in Cisco IOS Release 12.4(22)T. XML-PI provides a programmable interface which encapsulates IOS command-line interface (CLI) show commands in XML format in a consistent way across different Cisco products. Customers using XML-PI will be able to parse IOS show command output from within Tcl scripts using well-known keywords instead of having to depend on the use of regular expression support to "screen-scrape" output.

The benefit of using the XML-PI command extensions is to facilitate the extraction of specific output information that is generated using a CLI **show** command. Most show commands return many fields within the output and currently a regular expression has to be used to extract specific information that may appear in the middle of a line. XML-PI support provides a set of Tcl library functions to facilitate the parsing of output from the IOS CLI format extension in the form of:

```

show
<
show-command
> | format
{

```

```
spec-file
}
```

where a spec-file is a concatenation of all Spec File Entries (SFE) for each **show** command currently supported. As part of the XML-PI project a default spec-file will be included in the IOS Release 12.4(22)T images. The default spec-file will have a small set of commands and the SFE for the commands will have a subset of the possible tags. If no spec-file is provided with the format command, the default spec-file is used.

For more general details about XML-PI, see the "XML-PI" module.



## CHAPTER 104

# EEM Context Library Command Extensions

All the Tcl context library command extensions belong to the `::cisco::eem` namespace.

- [context\\_retrieve](#), on page 2171
- [context\\_save](#), on page 2174

## context\_retrieve

Retrieves Tcl variable(s) identified by the given context name, and possibly the scalar variable name, the array variable name, and the array index. Retrieved information is automatically deleted.



**Note** Once saved information is retrieved, it is automatically deleted. If that information is needed by another policy, the policy that retrieves it (using the **context\_retrieve** command extension) should also save it again (using the **context\_save** command extension).

### Syntax

```
context_retrieve ctxt [var] [index_if_array]
```

### Arguments

|                |                                                                                                                      |
|----------------|----------------------------------------------------------------------------------------------------------------------|
| ctxt           | (Mandatory) Context name.                                                                                            |
| var            | (Optional) Scalar variable name or array variable name. Defaults to a null string if this argument is not specified. |
| index_if_array | (Optional) The array index.                                                                                          |



**Note** The `index_if_array` argument will be ignored when the `var` argument is a scalar variable.

If `var` is unspecified, retrieves the whole variable table saved in the context.

If `var` is specified and `index_if_array` is not specified, or if `index_if_array` is specified but `var` is a scalar variable, retrieves the value of `var`.

If `var` is specified, and `index_if_array` is specified, and `var` is an array variable, retrieves the value of the specified array element.

### Result String

Resets the Tcl global variables to the state that they were in when the save was performed.

### Set\_cerrno

- A string displaying `_cerrno`, `_cerr_sub_num`, `_cerr_sub_err`, `_cerr_posix_err`, `_cerr_str` due to `appl_reqinfo` error.
- Variable is not in the context.

### Sample Usage

The following examples show how to use the `context_save` and `context_retrieve` command extension functionality to save and retrieve data. The examples are shown in save and retrieve pairs.

#### Example 1: Save

If `var` is unspecified or if a pattern is specified, saves multiple variables to the context.

```
::cisco::eem::event_register_none
namespace import ::cisco::eem::*
namespace import ::cisco::lib::*
set testvara 123
set testvarb 345
set testvarc 789
if {[catch {context_save TESTCTX "testvar*"} errmsg]} {
 action_syslog msg "context_save failed: $errmsg"
} else {
 action_syslog msg "context_save succeeded"
}
```

#### Example 1: Retrieve

If `var` is unspecified, retrieves multiple variables from the context.

```
::cisco::eem::event_register_none
namespace import ::cisco::eem::*
namespace import ::cisco::lib::*

if {[catch {foreach {var value} [context_retrieve TESTCTX] {set $var $value}} errmsg]} {
 action_syslog msg "context_retrieve failed: $errmsg"
} else {
 action_syslog msg "context_retrieve succeeded"
}

if {[info exists testvara]} {
 action_syslog msg "testvara exists and is $testvara"
} else {
 action_syslog msg "testvara does not exist"
}

if {[info exists testvarb]} {
 action_syslog msg "testvarb exists and is $testvarb"
}
```

```

} else {
 action_syslog msg "testvarb does not exist"
}
if {[info exists testvarc]} {
 action_syslog msg "testvarc exists and is $testvarc"
} else {
 action_syslog msg "testvarc does not exist"
}

```

### Example 2: Save

If var is specified, saves the value of var.

```

::cisco::eem::event_register_none

namespace import ::cisco::eem::*
namespace import ::cisco::lib::*
set testvar 123
if {[catch {context_save TESTCTX testvar} errmsg]} {
 action_syslog msg "context_save failed: $errmsg"
} else {
 action_syslog msg "context_save succeeded"
}

```

### Example 2: Retrieve

If var is specified and index\_if\_array is not specified, or if index\_if\_array is specified but var is a scalar variable, retrieves the value of var.

```

::cisco::eem::event_register_none
namespace import ::cisco::eem::*
namespace import ::cisco::lib::*
if {[catch {set testvar [context_retrieve TESTCTX testvar]} errmsg]} {
 action_syslog msg "context_retrieve failed: $errmsg"
} else {
 action_syslog msg "context_retrieve succeeded"
}
if {[info exists testvar]} {
 action_syslog msg "testvar exists and is $testvar"
} else {
 action_syslog msg "testvar does not exist"
}

```

### Example 3: Save

If var is specified, saves the value of var even if it is an array.

```

::cisco::eem::event_register_none

namespace import ::cisco::eem::*
namespace import ::cisco::lib::*
array set testvar "testvar1 ok testvar2 not_ok"
if {[catch {context_save TESTCTX testvar} errmsg]} {
 action_syslog msg "context_save failed: $errmsg"
} else {
 action_syslog msg "context_save succeeded"
}

```

**Example 3: Retrieve**

If `var` is specified, and `index_if_array` is not specified, and `var` is an array variable, retrieves the entire array.

```
::cisco::eem::event_register_none
namespace import ::cisco::eem::*
namespace import ::cisco::lib::*
if {[catch {array set testvar [context_retrieve TESTCTX testvar]} errmsg]} {
 action_syslog msg "context_retrieve failed: $errmsg"
} else {
 action_syslog msg "context_retrieve succeeded"
}
if {[info exists testvar]} {
 action_syslog msg "testvar exists and is [array get testvar]"
} else {
 action_syslog msg "testvar does not exist"
}
```

**Example 4: Save**

If `var` is specified, saves the value of `var` even if it is an array.

```
::cisco::eem::event_register_none
namespace import ::cisco::eem::*
namespace import ::cisco::lib::*
array set testvar "testvar1 ok testvar2 not_ok"
if {[catch {context_save TESTCTX testvar} errmsg]} {
 action_syslog msg "context_save failed: $errmsg"
} else {
 action_syslog msg "context_save succeeded"
}
```

**Example 4: Retrieve**

If `var` is specified, and `index_if_array` is specified, and `var` is an array variable, retrieves the specified array element value.

```
::cisco::eem::event_register_none
namespace import ::cisco::eem::*
namespace import ::cisco::lib::*
if {[catch {set testvar [context_retrieve TESTCTX testvar testvar1]} errmsg]} {
 action_syslog msg "context_retrieve failed: $errmsg"
} else {
 action_syslog msg "context_retrieve succeeded"
}
if {[info exists testvar]} {
 action_syslog msg "testvar exists and is $testvar"
} else {
 action_syslog msg "testvar doesn't exist"
}
```

## context\_save

Saves Tcl variables that match a given pattern in current and global namespaces with the given context name as identification. Use this Tcl command extension to save information outside of a policy. Saved information can be retrieved by a different policy using the **context\_retrieve** command extension.





**Note** Once saved information is retrieved, it is automatically deleted. If that information is needed by another policy, the policy that retrieves it (using the **context\_retrieve** command extension) should also save it again (using the **context\_save** command extension).

### Syntax

```
context_save ctxt [pattern]
```

### Arguments

|         |                                                                                                                                                                                                                                                                                                                                                                             |
|---------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| ctxt    | (Mandatory) Context name.                                                                                                                                                                                                                                                                                                                                                   |
| pattern | (Optional) The glob-style pattern as used by the <b>string match</b> Tcl command. If this argument is not specified, the pattern defaults to the wildcard *.<br><br>There are three constructs used in glob patterns: <ul style="list-style-type: none"> <li>• * = all characters</li> <li>• ? = 1 character</li> <li>• [abc] = match one of a set of characters</li> </ul> |

### Result String

None

### Set \_cerrno

A string displaying \_cerrno, \_cerr\_sub\_num, \_cerr\_sub\_err, \_cerr\_posix\_err, \_cerr\_str due to appl\_setinfo error.

### Sample Usage

The following examples show how to use the **context\_save** and **context\_retrieve** command extension functionality to save and retrieve data. The examples are shown in save and retrieve pairs.

#### Example 1: Save

If var is unspecified or if a pattern is specified, saves multiple variables to the context.

```
::cisco::eem::event_register_none
namespace import ::cisco::eem::*
namespace import ::cisco::lib::*
set testvara 123
set testvarb 345
set testvarc 789
if {[catch {context_save TESTCTX "testvar*"} errmsg]} {
 action_syslog msg "context_save failed: $errmsg"
} else {
 action_syslog msg "context_save succeeded"
}
```

**Example 1: Retrieve**

If var is unspecified, retrieves multiple variables from the context.

```

::cisco::eem::event_register_none
namespace import ::cisco::eem::*
namespace import ::cisco::lib::*

if {[catch {foreach {var value} [context_retrieve TESTCTX] {set $var $value}} errmsg]} {
 action_syslog msg "context_retrieve failed: $errmsg"
} else {
 action_syslog msg "context_retrieve succeeded"
}
if {[info exists testvara]} {
 action_syslog msg "testvara exists and is $testvara"
} else {
 action_syslog msg "testvara does not exist"
}
if {[info exists testvarb]} {
 action_syslog msg "testvarb exists and is $testvarb"
} else {
 action_syslog msg "testvarb does not exist"
}
if {[info exists testvarc]} {
 action_syslog msg "testvarc exists and is $testvarc"
} else {
 action_syslog msg "testvarc does not exist"
}
}

```

**Example 2: Save**

If var is specified, saves the value of var.

```

::cisco::eem::event_register_none

namespace import ::cisco::eem::*
namespace import ::cisco::lib::*
set testvar 123
if {[catch {context_save TESTCTX testvar} errmsg]} {
 action_syslog msg "context_save failed: $errmsg"
} else {
 action_syslog msg "context_save succeeded"
}
}

```

**Example 2: Retrieve**

If var is specified and index\_if\_array is not specified, or if index\_if\_array is specified but var is a scalar variable, retrieves the value of var.

```

::cisco::eem::event_register_none
namespace import ::cisco::eem::*
namespace import ::cisco::lib::*
if {[catch {set testvar [context_retrieve TESTCTX testvar]} errmsg]} {
 action_syslog msg "context_retrieve failed: $errmsg"
} else {
 action_syslog msg "context_retrieve succeeded"
}
if {[info exists testvar]} {
 action_syslog msg "testvar exists and is $testvar"
} else {
}
}

```

```

 action_syslog msg "testvar does not exist"
 }

```

### Example 3: Save

If var is specified, saves the value of var even if it is an array.

```

::cisco::eem::event_register_none

namespace import ::cisco::eem::*
namespace import ::cisco::lib::*
array set testvar "testvar1 ok testvar2 not_ok"
if {[catch {context_save TESTCTX testvar} errmsg]} {
 action_syslog msg "context_save failed: $errmsg"
} else {
 action_syslog msg "context_save succeeded"
}

```

### Example 3: Retrieve

If var is specified, and index\_if\_array is not specified, and var is an array variable, retrieves the entire array.

```

::cisco::eem::event_register_none
namespace import ::cisco::eem::*
namespace import ::cisco::lib::*
if {[catch {array set testvar [context_retrieve TESTCTX testvar]} errmsg]} {
 action_syslog msg "context_retrieve failed: $errmsg"
} else {
 action_syslog msg "context_retrieve succeeded"
}
if {[info exists testvar]} {
 action_syslog msg "testvar exists and is [array get testvar]"
} else {
 action_syslog msg "testvar does not exist"
}

```

### Example 4: Save

If var is specified, saves the value of var even if it is an array.

```

::cisco::eem::event_register_none
namespace import ::cisco::eem::*
namespace import ::cisco::lib::*
array set testvar "testvar1 ok testvar2 not_ok"
if {[catch {context_save TESTCTX testvar} errmsg]} {
 action_syslog msg "context_save failed: $errmsg"
} else {
 action_syslog msg "context_save succeeded"
}

```

### Example 4: Retrieve

If var is specified, and index\_if\_array is specified, and var is an array variable, retrieves the specified array element value.

```

::cisco::eem::event_register_none
namespace import ::cisco::eem::*
namespace import ::cisco::lib::*
if {[catch {set testvar [context_retrieve TESTCTX testvar testvar1]} errmsg]} {

```

```
 action_syslog msg "context_retrieve failed: $errmsg"
 } else {
 action_syslog msg "context_retrieve succeeded"
 }
 if {[info exists testvar]} {
 action_syslog msg "testvar exists and is $testvar"
 } else {
 action_syslog msg "testvar doesn't exist"
 }
}
```



## CHAPTER 105

# EEM Event Registration Tcl Command Extensions

The following conventions are used for the syntax documented on the Tcl command extension pages:

- An optional argument is shown within square brackets, for example:

[type ?]

- A question mark ? represents a variable to be entered.
- Choices between arguments are represented by pipes, for example:

priority low|normal|high



---

**Note** For all EEM Tcl command extensions, if there is an error, the returned Tcl result string contains the error information.

---



---

**Note** Arguments for which no numeric range is specified take an integer from -2147483648 to 2147483647, inclusive.

---

- [event\\_register\\_appl](#), on page 2180
- [event\\_register\\_cli](#), on page 2182
- [event\\_register\\_counter](#), on page 2185
- [event\\_register\\_gold](#), on page 2187
- [event\\_register\\_identity](#), on page 2193
- [event\\_register\\_interface](#), on page 2195
- [event\\_register\\_ioswdsysmon](#), on page 2200
- [event\\_register\\_ipsla](#), on page 2203
- [event\\_register\\_mat](#), on page 2206
- [event\\_register\\_neighbor\\_discovery](#), on page 2208
- [event\\_register\\_nf](#), on page 2211
- [event\\_register\\_none](#), on page 2214
- [event\\_register\\_oir](#), on page 2216
- [event\\_register\\_process](#), on page 2218
- [event\\_register\\_resource](#), on page 2220
- [event\\_register\\_rf](#), on page 2222

- [event\\_register\\_routing](#), on page 2225
- [event\\_register\\_rpc](#), on page 2227
- [event\\_register\\_snmp](#), on page 2229
- [event\\_register\\_snmp\\_notification](#), on page 2233
- [event\\_register\\_snmp\\_object](#), on page 2235
- [event\\_register\\_syslog](#), on page 2238
- [event\\_register\\_timer](#), on page 2240
- [event\\_register\\_timer\\_subscriber](#), on page 2244
- [event\\_register\\_track](#), on page 2246
- [event\\_register\\_wdsysmon](#), on page 2248

## event\_register\_appl

Registers for an application event. Use this Tcl command extension to run a policy when an application event is triggered following another policy's execution of an **event\_publish** Tcl command extension; the **event\_publish** command extension publishes an application event.

In order to register for an application event, a subsystem must be specified. Either a Tcl policy or the internal Embedded Event Manager (EEM) API can publish an application event. If the event is being published by a policy, the sub\_system argument that is reserved for a policy is 798.

### Syntax

```
event_register_appl [tag ?] sub_system ? type ? [queue_priority low|normal|high|last] [maxrun
?] [nice 0|1]
```

### Arguments

|            |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |
|------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| tag        | (Optional) String identifying a tag that can be used with the trigger Tcl command extension to support multiple event statements within a Tcl script.                                                                                                                                                                                                                                                                                                                                                    |
| sub_system | (Mandatory) Number assigned to the EEM policy that published the application event. The number is set to 798 because all other numbers are reserved for Cisco use. If this argument is not specified, all components are matched.                                                                                                                                                                                                                                                                        |
| type       | (Mandatory) Event subtype within the specified event. The sub_system and type arguments uniquely identify an application event. If this argument is not specified, all types are matched. If you specify this argument, you must choose an integer between 1 and 4294967295, inclusive.<br><br>There must be a match of component and type between the <b>event_publish</b> command extension and the <b>event_register_appl</b> command extension in order for the publishing and registration to work. |

|                |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
|----------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| queue_priority | <p>(Optional) Priority level at which the script will be queued:</p> <ul style="list-style-type: none"> <li>• queue_priority low--Specifies that the script is to be queued at the lowest of the three priority levels.</li> <li>• queue_priority normal--Specifies that the script is to be queued at a priority level greater than low priority but less than high priority.</li> <li>• queue_priority high--Specifies that the script is to be queued at the highest of the three priority levels.</li> <li>• queue_priority last--Specifies that the script is to be queued at the lowest priority level.</li> </ul> <p>If more than one script is registered with the "queue_priority_last" argument set, these scripts will execute in the order in which the events are published.</p> <p><b>Note</b> The queue_priority argument specifies the queuing priority, but not the execution priority, of the script being registered.</p> <p>If this argument is not specified, the default queuing priority is normal.</p> |
| maxrun         | <p>(Optional) Maximum run time of the script (specified in SSSSSSSSS[.MMM] format, where SSSSSSSSS must be an integer representing seconds between 0 and 4294967295, inclusive, and where MMM must be an integer representing milliseconds between 0 and 999). If this argument is not specified, the default 20-second run-time limit is used.</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |
| nice           | <p>(Optional) Policy run-time priority setting. When the nice argument is set to 1, the policy is run at a run-time priority that is less than the default priority. The default value is 0.</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |

If multiple conditions exist, the application event will be raised when all the conditions are satisfied.

**Result String**

None

**Set\_cerrno**

No

Event\_reqinfo

**Event\_reqinfo**

```
"event_id %u event_type %u event_type_string {%s} event_pub_sec %u event_pub_msec %u"
"sub_system 0x%x type %u data1 {%s} data2 {%s} data3 {%s} data4 {%s}"
```

| Event Type        | Description                                                                                                                                                 |
|-------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------|
| event_id          | Unique number that indicates the ID for this published event. Multiple policies may be run for the same event, and each policy will have the same event_id. |
| event_type        | Type of event.                                                                                                                                              |
| event_type_string | An ASCII string that represents the name of the event for this event type.                                                                                  |

| Event Type                                    | Description                                                                                                                                                                   |
|-----------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>event_pub_sec</b><br><b>event_pub_msec</b> | The time, in seconds and milliseconds, when the event was published to the Embedded Event Manager (EEM).                                                                      |
| <b>sub_system</b>                             | Number assigned to the EEM policy that published the application event. Number is set to 798 because all other numbers are reserved for Cisco use.                            |
| <b>type</b>                                   | Event subtype within the specified component.                                                                                                                                 |
| <b>data1 data2 data3 data4</b>                | Argument data that is passed to the application-specific event when the event is published. The data is character text, an environment variable, or a combination of the two. |

## event\_register\_cli

Registers for a CLI event. Use this Tcl command extension to run a policy when a CLI command of a specific pattern is entered based on pattern matching performed against an expanded CLI command.



**Note** The user can enter an abbreviated CLI command, such as **sh mem summary**, and the parser will expand the command to **show memory summary** to perform the matching.



**Note** The functionality provided in the CLI event detector only allows a regular expression pattern match on a valid IOS CLI command itself. This does not include text after a pipe character when redirection is used.

### Syntax

```
event_register_cli [tag ?] sync yes|no skip yes|no
[occurs ?] [period ?] pattern ? [default ?] [enter] [questionmark] [tab] [mode]
[queue_priority low|normal|high|last] [maxrun ?] [nice 0|1]
```

### Arguments

|      |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
|------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| tag  | (Optional) String identifying a tag that can be used with the trigger Tcl command extension to support multiple event statements within a Tcl script.                                                                                                                                                                                                                                                                                                                                                                                        |
| sync | (Mandatory) A "yes" means that the policy (the event publish) will run synchronously with the CLI command; a "no" means that the event publish will be performed asynchronously with the CLI command. The event detector will be notified when the policy completes running. The exit status of the policy indicates whether or not the CLI command should be executed: if the exit status is zero, which means that the policy is executed successfully, the CLI command will not be executed; otherwise, the CLI command will be executed. |



|                |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
|----------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| skip           | <p>Mandatory if the sync argument is "no" and should not exist if the sync argument is "yes." If the skip argument is "yes," it means that the CLI command should not be executed. If the skip argument is "no," it means that the CLI command should be executed.</p> <p><b>Caution</b> When the skip argument is "yes," unintended results may be produced if the pattern match is made for configuration commands because the CLI command that matches the regular expression will not be executed.</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |
| occurs         | <p>(Optional) The number of occurrences before the event is raised. If this argument is not specified, the event is raised on the first occurrence. If this argument is specified, it must be an integer between 1 and 4294967295, inclusive.</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
| period         | <p>(Optional) Specifies a backward looking time window in which all CLI events must occur (the occurs clause must be satisfied) in order for an event to be published (specified in SSSSSSSSS[.MMM] format, where SSSSSSSSS must be an integer representing seconds between 0 and 4294967295, inclusive, and where MMM must be an integer representing milliseconds between 0 and 999). If this argument is not specified, the most recent event is used.</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |
| pattern        | <p>(Mandatory) Specifies the regular expression used to perform the CLI command pattern match.</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |
| default        | <p>(Optional) The time period during which the CLI event detector waits for the policy to exit (specified in SSSSSSSSS[.MMM] format, where SSSSSSSSS must be an integer representing seconds between 0 and 4294967295, inclusive, and where MMM must be an integer representing milliseconds between 0 and 999). If the default time period expires before the policy exits, the default action will be executed. The default action is to run the command. If this argument is not specified, the default time period is set to 30 seconds.</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |
| queue_priority | <p>(Optional) Priority level at which the script will be queued:</p> <ul style="list-style-type: none"> <li>• queue_priority low--Specifies that the script is to be queued at the lowest of the three priority levels.</li> <li>• queue_priority normal--Specifies that the script is to be queued at a priority level greater than low priority but less than high priority.</li> <li>• queue_priority high--Specifies that the script is to be queued at the highest of the three priority levels.</li> <li>• queue_priority last--Specifies that the script is to be queued at the lowest priority level.</li> </ul> <p>If more than one script is registered with the "queue_priority_last" argument set, these scripts will execute in the order in which the events are published.</p> <p><b>Note</b> The queue_priority argument specifies the queuing priority, but not the execution priority, of the script being registered.</p> <p>If this argument is not specified, the default queuing priority is normal.</p> |
| enter          | <p>(Optional) Specifies to perform the event match when the user presses the Enter key. When this parameter is used, the input string will not be expanded before matching.</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
| questionmark   | <p>(Optional) Specifies to perform the event match when the user presses the ? key. When this parameter is used, the input string will not be expanded before matching.</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |

|        |                                                                                                                                                                                                                                                                                                                                              |
|--------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| tab    | (Optional) Specifies to perform the event match when the user presses the Tab key. When this parameter is used, the input string will not be expanded before matching.                                                                                                                                                                       |
| mode   | (Optional) Events will only be generated when the parser is in the specified parser mode. The available modes can be listed using the <b>show parser dump</b> CLI command. The mode parameter is checked when any one of the optional parameters--enter, questionmark, or tab-- is specified.                                                |
| maxrun | (Optional) Maximum run time of the script (specified in SSSSSSSSS[.MMM] format, where SSSSSSSSS must be an integer representing seconds between 0 and 4294967295, inclusive, and where MMM must be an integer representing milliseconds between 0 and 999). If this argument is not specified, the default 20-second run-time limit is used. |
| nice   | (Optional) Policy run-time priority setting. When the nice argument is set to 1, the policy is run at a run-time priority that is less than the default priority. The default value is 0.                                                                                                                                                    |

If multiple conditions are specified, the CLI event will be raised when all the conditions are matched.

### Result String

None

### Set\_cerrno

No



**Note** This policy runs before the CLI command is executed. For example, suppose policy\_CLI is registered to run when the **copy** command is entered. When the **copy** command is entered, the CLI event detector finds a pattern match and triggers this policy to run. When the policy execution ends, the CLI event detector determines if the **copy** command needs to be executed according to "sync", "skip" (set in the policy), and the exit status of the policy execution if needed.

### Event\_reqinfo

```
"event_id %u event_type %u event_type_string {%s} event_pub_sec %u event_pub_msec %u
event_severity %u msg {%s} msg_count %d line %u key %u tty %u error_code %u"
```

| Event Type                      | Description                                                                                                                                                 |
|---------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------|
| event_id                        | Unique number that indicates the ID for this published event. Multiple policies may be run for the same event, and each policy will have the same event_id. |
| event_type                      | Type of event.                                                                                                                                              |
| event_type_string               | An ASCII string that represents the name of the event for this event type.                                                                                  |
| event_pub_sec<br>event_pub_msec | The time, in seconds and milliseconds, at which the event was published to the EEM.                                                                         |
| event_severity                  | The severity of the event.                                                                                                                                  |

| Event Type | Description                                                                                                                                                                                                        |
|------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| msg        | Text entered at the CLI prompt.                                                                                                                                                                                    |
| msg_count  | Number of times the pattern matched before the event was triggered.                                                                                                                                                |
| line       | The text the parser was able to expand up to the point where the matched key was entered.                                                                                                                          |
| key        | The enter, questionmark, or tab key.                                                                                                                                                                               |
| tty        | Corresponds to the line number the user is executing the command on.                                                                                                                                               |
| error_code | The error code in CLI.<br>0 --No error from parser up to point where a key was entered.<br>1--Command is ambiguous up to point where a key was entered.<br>4--Unknown command up to point where a key was entered. |

## event\_register\_counter

Registers for a counter event as both a publisher and a subscriber. Use this Tcl command extension to run a policy on the basis of a named counter crossing a threshold. This event counter, as a subscriber, identifies the name of the counter to which it wants to subscribe and depends on another policy or another process to actually manipulate the counter. For example, let policyB act as a counter policy, whereas policyA (although it does not need to be a counter policy) uses **register\_counter**, **counter\_modify**, or **unregister\_counter** Tcl command extensions to manipulate the counter defined in policyB.

### Syntax

```
event_register_counter [tag ?] name ? entry_op gt|ge|eq|ne|lt|le entry_val ?
exit_op gt|ge|eq|ne|lt|le exit_val ? [queue_priority low|normal|high|last]
[maxrun ?] [nice 0|1]
```

### Arguments

|           |                                                                                                                                                                                                           |
|-----------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| tag       | (Optional) String identifying a tag that can be used with the trigger Tcl command extension to support multiple event statements within a Tcl script.                                                     |
| name      | (Mandatory) Name of the counter.                                                                                                                                                                          |
| entry_op  | (Mandatory) Entry comparison operator used to compare the current counter value with the entry value; if true, an event will be raised and event monitoring will be disabled until exit criteria are met. |
| entry_val | (Mandatory) Value with which the current counter value should be compared to decide if the counter event should be raised.                                                                                |
| exit_op   | (Mandatory) Exit comparison operator used to compare the current counter value with the exit value; if true, event monitoring for this event will be reenabled.                                           |

|                |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
|----------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| exit_val       | (Mandatory) Value with which the current counter value should be compared to decide if the exit criteria are met.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
| queue_priority | <p>(Optional) Priority level at which the script will be queued:</p> <ul style="list-style-type: none"> <li>• queue_priority low--Specifies that the script is to be queued at the lowest of the three priority levels.</li> <li>• queue_priority normal--Specifies that the script is to be queued at a priority level greater than low priority but less than high priority.</li> <li>• queue_priority high--Specifies that the script is to be queued at the highest of the three priority levels.</li> <li>• queue_priority last--Specifies that the script is to be queued at the lowest priority level.</li> </ul> <p>If more than one script is registered with the "queue_priority_last" argument set, these scripts will execute in the order in which the events are published.</p> <p><b>Note</b> The queue_priority argument specifies the queuing priority, but not the execution priority, of the script being registered.</p> <p>If this argument is not specified, the default queuing priority is normal.</p> |
| maxrun         | (Optional) Maximum run time of the script (specified in SSSSSSSSS[.MMM] format, where SSSSSSSSS must be an integer representing seconds between 0 and 4294967295, inclusive, and where MMM must be an integer representing milliseconds between 0 and 999). If this argument is not specified, the default 20-second run-time limit is used.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |
| nice           | (Optional) Policy run-time priority setting. When the nice argument is set to 1, the policy is run at a run-time priority that is less than the default priority. The default value is 0.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |

**Result String**

None

**Set\_cerrno**

No

**Event\_reqinfo**

```
"event_id %u event_type %u event_type_string {%s} %u event_pub_sec %u event_pub_msec %u"
"name {%s}"
```

| Event Type | Description                                                                                                                                                 |
|------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------|
| event_id   | Unique number that indicates the ID for this published event. Multiple policies may be run for the same event, and each policy will have the same event_id. |
| event_type | Type of event.                                                                                                                                              |

| Event Type                   | Description                                                                     |
|------------------------------|---------------------------------------------------------------------------------|
| event_type_string            | An ASCII string that represents the name of the event for this event type.      |
| event_pub_sec event_pub_msec | The time, in seconds and milliseconds, when the event was published to the EEM. |
| name                         | Counter name.                                                                   |

## event\_register\_gold

Registers for a Generic Online Diagnostic (GOLD) failure event. Use this Tcl command extension to run a policy on the basis of a Generic Online Diagnostic (GOLD) failure event for the specified card and subcard.

### Syntax

```
event_register_gold card all|card_number
[subcard all|subcard_number]
[new_failure TRUE|FALSE]
[severity_major TRUE]
[severity_minor TRUE]
[severity_normal TRUE]
[action_notify TRUE|FALSE]
[testing_type [bootup|ondemand|schedule|monitoring]]
[test_name [testname]]
[test_id [testnumber]]
[consecutive_failure consecutive_failure_number]
[platform_action [action_flag]]
[maxrun ?]
[queue_priority low|normal|high|last]
[nice 0|1]
```

### Arguments

|         |                                                                                                                                                                                                                                                                                                                                                                                                                                  |
|---------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| card    | <p>(Mandatory) Specifies whether all cards or one card is to be monitored:</p> <ul style="list-style-type: none"> <li>• card all--Specifies that all cards are to be monitored. This is the default.</li> <li>• card-number--Specifies that the card identified by the number card-number is to be monitored.</li> </ul> <p>This argument must be specified to complete the <b>event_register_gold</b>Tcl command extension.</p> |
| subcard | <p>(Optional) Specifies that one or more subcards are to be monitored:</p> <ul style="list-style-type: none"> <li>• subcard all--Specifies that all subcards are to be monitored.</li> <li>• subcard-number--Specifies that the subcard identified by the number subcard-number is to be monitored.</li> </ul> <p>If this argument is not specified, all subcards are monitored by default.</p>                                  |

|                 |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |
|-----------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| new_failure     | <p>(Optional) Specifies event criteria based on the new test failure information from GOLD:</p> <ul style="list-style-type: none"> <li>• new_failure TRUE--Specifies that the event criterion for the new test failure is true from GOLD.</li> <li>• new_failure FALSE--Specifies that the event criterion for the new test failure is false from GOLD.</li> </ul> <p>If this argument is not specified, the new test failure information from GOLD is not considered in the event criteria.</p>                                                                                                                                                                                                                                                                                                |
| severity_major  | <p>(Optional) Specifies that the event criteria for diagnostic result matches with the diagnostic major error from GOLD.</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |
| severity_minor  | <p>(Optional) Specifies that the event criteria for diagnostic result matches with diagnostic minor error from GOLD.</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |
| severity_normal | <p>(Optional) Specifies that the event criteria for diagnostic result matches with diagnostic normal from GOLD. This is the default.</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |
| action_notify   | <p>(Optional) Specifies the event criteria based on the action notify information from GOLD:</p> <ul style="list-style-type: none"> <li>• action_notify TRUE--Specifies that the event criterion for the action notify is true from GOLD.</li> <li>• action_notify FALSE--Specifies that the event criterion for the action notify is false from GOLD.</li> </ul> <p>If this argument is not specified, the action notify information from GOLD is not considered in the event criteria.</p>                                                                                                                                                                                                                                                                                                    |
| testing_type    | <p>(Optional) Specifies the event criteria based on the testing types of the diagnostic from GOLD:</p> <ul style="list-style-type: none"> <li>• testing_type bootup--Specifies the diagnostic tests that are running on system bootup.</li> <li>• testing_type ondemand--Specifies the diagnostic tests that are running from CLI after the card is online.</li> <li>• testing_type schedule--Specifies the scheduled diagnostic tests.</li> <li>• testing_type monitoring--Specifies the diagnostic tests that are running periodically in the background to monitor the health of the system.</li> </ul> <p>If this argument is not specified, the testing type information from GOLD is not considered in the event criteria and the policy applies to all the diagnostic testing types.</p> |
| test_name       | <p>(Optional) Specifies the event criteria based on the test name:</p> <ul style="list-style-type: none"> <li>• test_name test-name--Specifies the event criteria based on the test with the name test-name.</li> </ul> <p>If this argument is not specified, the test name information from GOLD is not considered in the event criteria.</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                  |

|                     |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |
|---------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| test_id             | <p>(Optional) Specifies the event criteria based on test ID:</p> <ul style="list-style-type: none"> <li>test_id test-id--Specifies the event criteria based on the test with the ID number test-id. The maximum value of test-id is 65535.</li> </ul> <p><b>Note</b> Because the test ID can be different for the same test on different line cards, usually the test_name keyword should be used instead. If the test ID is specified and conflicts with the specified test name, the test name overwrites the test ID.</p> <p>If this argument is not specified, test ID information from GOLD is not considered in the event criteria.</p>                                              |
| consecutive_failure | <p>(Optional) Specifies the event criteria based on consecutive test failure information from GOLD:</p> <ul style="list-style-type: none"> <li>consecutive_failure consecutive-failure-number--Specifies that the event criterion is based on the occurrence of consecutive-failure-number consecutive test failures.</li> </ul> <p>If this argument is not specified, consecutive test failure information from GOLD is not considered in the event criteria.</p>                                                                                                                                                                                                                         |
| platform_action     | <p>(Optional) Specifies whether callback to the platform is needed when all the event criteria are matched. When callback is needed, the platform needs to register a callback function through the provided registry.</p> <ul style="list-style-type: none"> <li>platform_action action-flag-number--Specifies that, when callback to the platform is needed, specific information is specified by the platform-specific action-flag-number value. The maximum value of action-flag-number is 65535.</li> </ul> <p><b>Note</b> It is up to the platform to determine what action needs to be taken based on the flag.</p> <p>If this argument is not specified, there is no callback.</p> |
| maxrun              | <p>(Optional) Specifies the maximum runt time of the script.</p> <ul style="list-style-type: none"> <li>maxrun max-run-time-number--Specifies that the maximum run time of the script is max-run-time-number seconds. The maximum value of max-run-time-number is 4294967295 seconds.</li> </ul> <p>If this argument is not specified, the default run time is 20 seconds.</p>                                                                                                                                                                                                                                                                                                             |

|                |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
|----------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| queue_priority | <p>(Optional) Priority level at which the script will be queued:</p> <ul style="list-style-type: none"> <li>• queue_priority low--Specifies that the script is to be queued at the lowest of the three priority levels.</li> <li>• queue_priority normal--Specifies that the script is to be queued at a priority level greater than low priority but less than high priority.</li> <li>• queue_priority high--Specifies that the script is to be queued at the highest of the three priority levels.</li> <li>• queue_priority last--Specifies that the script is to be queued at the lowest priority level.</li> </ul> <p>If more than one script is registered with the "queue_priority_last" argument set, these scripts will execute in the order in which the events are published.</p> <p><b>Note</b> The queue_priority argument specifies the queuing priority, but not the execution priority, of the script being registered.</p> <p>If this argument is not specified, the default queuing priority is normal.</p> |
| nice           | <p>(Optional) Policy run-time priority setting:</p> <ul style="list-style-type: none"> <li>• nice 0--Specifies that the policy is run at the default run-time priority level.</li> <li>• nice 1--Specifies that the policy is run at a run-time priority that is less than the default priority.</li> </ul> <p>If this argument is not specified, the default run-time priority is used.</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |

**Result String**

None

**Set\_cerrno**

No

**Event\_reqinfo**

```
"event_id %u event_type %u event_type_string {%s} %u card %u sub_card %u"
"event_severity {%s} event_pub_sec %u event_pub_msec %u overall_result %u"
"new_failure {%s} action_notify {%s} tt %u tc %u bl %u ci %u pc %u cn {%s}"
"sn {%s} tn# {%s} ta# %s ec# {%s} rc# %u lf# {%s} tf# %u cf# %u tr# {%s}"
"tr#p# {%s} tr#d# {%s}"
```

| Event Type    | Description                                             |
|---------------|---------------------------------------------------------|
| action_notify | Action notify information in GOLD event: true or false. |



| Event Type                                    | Description                                                                                                                                                                                                                                     |
|-----------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>bl</b>                                     | The boot-up diagnostic level, which can be one of the following values: <ul style="list-style-type: none"> <li>• 0: complete diagnostic</li> <li>• 1: minimal diagnostics</li> <li>• 2: bypass diagnostic</li> </ul>                            |
| <b>card</b>                                   | Card information for the GOLD event.                                                                                                                                                                                                            |
| <b>cf</b> <i>testnum</i>                      | Consecutive failure, where <i>testnum</i> is the test number. For example, <b>cf3</b> is the EEM built-in environment variable for consecutive failure of test 3.                                                                               |
| <b>ci</b>                                     | Card index.                                                                                                                                                                                                                                     |
| <b>cn</b>                                     | Card name.                                                                                                                                                                                                                                      |
| <b>ec</b> <i>testnum</i>                      | Test error code, where <i>testnum</i> is the test number. For example, <b>ec3</b> is the EEM built-in environment variable for the error code of test 3.                                                                                        |
| <b>event_id</b>                               | Unique number that indicates the ID for this published event. Multiple policies may be run for the same event, and each policy will have the same event_id.                                                                                     |
| <b>event_pub_msec</b><br><b>event_pub_sec</b> | The time, in milliseconds and seconds, when the event was published to the EEM.                                                                                                                                                                 |
| <b>event_severity</b>                         | GOLD event severity, which can be one of the following values: <ul style="list-style-type: none"> <li>• normal</li> <li>• minor</li> <li>• major.</li> </ul>                                                                                    |
| <b>event_type</b>                             | Type of event.                                                                                                                                                                                                                                  |
| <b>event_type_string</b>                      | An ASCII string that represents the name of the event for this event type.                                                                                                                                                                      |
| <b>lf</b> <i>testnum</i>                      | Last fail time, where <i>testnum</i> is the test number. For example, <b>lf3</b> is the EEM built-in variable for the last fail time of test 3.<br><br>The timestamp format is <i>mmm dd yyyy hh:mm:ss</i> . For example, Mar 11 1960 08:47:00. |
| <b>new_failure</b>                            | The new test failure information in a GOLD event flag: true or false.                                                                                                                                                                           |
| <b>overall_result</b>                         | The overall diagnostic result, which can be one of the following values: <ul style="list-style-type: none"> <li>• 0: OK</li> <li>• 3: minor error</li> <li>• 4: major error</li> <li>• 14: unknown result</li> </ul>                            |

| Event Type                                       | Description                                                                                                                                                                                                                                                                                                                                                                                                                  |
|--------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>pc</b>                                        | Port counts.                                                                                                                                                                                                                                                                                                                                                                                                                 |
| <b>rc</b> <i>testnum</i>                         | Test total run count, where <i>testnum</i> is the test number. For example, <b>rc3</b> is the EEM built-in variable for the total run count of test 3.                                                                                                                                                                                                                                                                       |
| <b>sn</b>                                        | Card serial number.                                                                                                                                                                                                                                                                                                                                                                                                          |
| <b>sub_card</b>                                  | The subcard on which a GOLD failure event was detected.                                                                                                                                                                                                                                                                                                                                                                      |
| <b>ta</b> <i>testnum</i>                         | Test attribute, where <i>testnum</i> is the test number. For example, <b>ta3</b> is the EEM built-in variable for the test attribute of test 3.                                                                                                                                                                                                                                                                              |
| <b>tc</b>                                        | Test counts.                                                                                                                                                                                                                                                                                                                                                                                                                 |
| <b>tf</b> <i>testnum</i>                         | Total failure count, where <i>testnum</i> is the test number. For example, <b>tf3</b> is the EEM built-in variable for the total failure count of test 3.                                                                                                                                                                                                                                                                    |
| <b>tn</b> <i>testnum</i>                         | Test name, where <i>testnum</i> is the test number. For example, <b>tn3</b> is the EEM built-in variable for the name of test 3.                                                                                                                                                                                                                                                                                             |
| <b>tr</b> <i>testnum</i>                         | Test result, where <i>testnum</i> is the test number. For example, <b>tr6</b> is the EEM built-in variable for test 6 where test 6 is not a per-port test and not a per-device test.<br><br>The test result is one of the following values: <ul style="list-style-type: none"> <li>• P: diagnostic result Pass</li> <li>• F: diagnostic result Fail</li> <li>• U: diagnostic result Unknown</li> </ul>                       |
| <b>tr</b> <i>testnum</i> <b>d</b> <i>devnum</i>  | Per-device test result, where <i>testnum</i> is the test number and <i>devnum</i> is the device number. For example, <b>tr3d20</b> is the EEM built-in variable for the test result for test 3, device 20.<br><br>The test result is one of the following values: <ul style="list-style-type: none"> <li>• P: diagnostic result Pass</li> <li>• F: diagnostic result Fail</li> <li>• U: diagnostic result Unknown</li> </ul> |
| <b>tr</b> <i>testnum</i> <b>p</b> <i>portnum</i> | Per-port test result, where <i>testnum</i> is the test number and <i>portnum</i> is the device number. For example, <b>tr5p20</b> is the EEM built-in variable for the test result for test 3, port 20.<br><br>The test result is one of the following values: <ul style="list-style-type: none"> <li>• P: diagnostic result Pass</li> <li>• F: diagnostic result Fail</li> <li>• U: diagnostic result Unknown</li> </ul>    |

| Event Type | Description                                                                                                                                                                                                                                      |
|------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| tt         | The testing type, which can be one of the following: <ul style="list-style-type: none"> <li>• 1: A boot-up diagnostic</li> <li>• 2: An on-demand diagnostic</li> <li>• 3: A schedule diagnostic</li> <li>• 4: A monitoring diagnostic</li> </ul> |

## event\_register\_identity

Registers for an identity event. Use this Tcl command extension to generate an event when AAA authentication or authorization is successful or failure or after normal user traffic on the port is allowed to flow.

### Syntax

```
event_register_identity [tag ?] interface ?
[aaa-attribute ?]
[authc {all | fail | success}]
[authz {all | fail | success}]
[authz-complete]
[mac-address ?]
[queue_priority {normal | low | high | last}]
[maxrun ?] [nice {0 | 1}]
```

### Arguments

|                |                                                                                                                                                                                                                                                                                                                                            |
|----------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| tag            | (Optional) String identifying a tag that can be used with the trigger Tcl command extension to support multiple event statements within a Tcl script.                                                                                                                                                                                      |
| interface      | A regular expression pattern to match against interface names.                                                                                                                                                                                                                                                                             |
| aaa-attribute  | (Optional) A regular expression that can be used to filter events by specific AAA attributes.                                                                                                                                                                                                                                              |
| authc          | (Optional) Triggers events on successful, failed or both successful and failed authentication.                                                                                                                                                                                                                                             |
| authz          | (Optional) Triggers events on successful, failed or both successful and failed authorization.                                                                                                                                                                                                                                              |
| authz-complete | (Optional) Triggers events once the device connected to the interface is fully authenticated, authorized and normal traffic has begun to flow on that interface.                                                                                                                                                                           |
| mac-address    | (Optional) A regular expression pattern that can be used to filter events by mac addresses of the remote device.                                                                                                                                                                                                                           |
| maxrun         | (Optional) Maximum run time of the script (specified in SSSSSSSSS[.MMM] format, where SSSSSSSSS must be an integer representing seconds between 0 and 31536000, inclusive, and where MMM must be an integer representing milliseconds between 0 and 999). If this argument is not specified, the default 20-second run-time limit is used. |

|                |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |
|----------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| queue_priority | <p>(Optional) Priority level at which the script will be queued:</p> <ul style="list-style-type: none"> <li>• queue_priority low--Specifies that the script is to be queued at the lowest of the three priority levels.</li> <li>• queue_priority normal--Specifies that the script is to be queued at a priority level greater than low priority but less than high priority.</li> <li>• queue_priority high--Specifies that the script is to be queued at the highest of the three priority levels.</li> <li>• queue_priority last--Specifies that the script is to be queued at the lowest priority level.</li> </ul> <p>If more than one script is registered with the "queue_priority_last" argument set, these scripts will execute in the order in which the events are published.</p> <p>The queue_priority argument specifies the queuing priority, but not the execution priority, of the script being registered.</p> <p>If this argument is not specified, the default queuing priority is normal.</p> |
| nice           | <p>(Optional) Policy run-time priority setting. When the nice argument is set to 1, the policy is run at a run-time priority that is less than the default priority. The default value is 0.</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |

**Result String**

None

**Set\_cerrno**

No

**Event\_reqinfo For EEM\_EVENT\_IDENTITY**

```
"event_id %u event_type %u event_type_string {%s} event_pub_sec %u event_pub_msec %u
event_severity %u identity_stage %u identity_status %u interface %u identity_mac %u
identity_attribute > {%s}"
```

| Event Type                          | Description                                                                                                                                                 |
|-------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>event_id</b>                     | Unique number that indicates the ID for this published event. Multiple policies may be run for the same event, and each policy will have the same event_id. |
| <b>event_type</b>                   | Type of event.                                                                                                                                              |
| <b>event_type_string</b>            | An ASCII string that represents the name of the event for this event type.                                                                                  |
| <b>event_pub_sec event_pub_msec</b> | The time, in seconds and milliseconds, at which the event was published to the EEM.                                                                         |
| <b>event_severity</b>               | The severity of the event.                                                                                                                                  |

| Event Type                        | Description                                                                                                                                                      |
|-----------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>identity_stage</b>             | One among authentication, authorization or authorization-complete stages.                                                                                        |
| <b>identity_status</b>            | Success or one of these failure types: fail_authc, fail_aaa_server, fail_no_response, fail_timeout, fail_authz. For authorization-complete it is always success. |
| <b>interface</b>                  | The interface for the event.                                                                                                                                     |
| <b>identity_mac</b>               | The MAC address of the remote device for the event.                                                                                                              |
| <b>identity_&lt;attribute&gt;</b> | For each AAA attribute, a set a dynamic variable to the value corresponding to that AAA attribute in the attribute or value list.                                |

## event\_register\_interface

Registers for an interface counter event. Use this Tcl command extension to generate an event when specified interface counters exceed specified thresholds.

### Syntax

```
event_register_interface [tag ?] name ?
parameter ? entry_op gt|ge|eq|ne|lt|le
entry_val ? entry_val_is_increment TRUE|FALSE
entry_type value|increment|rate
[exit_comb or|and]
[exit_op gt|ge|eq|ne|lt|le]
[exit_val ?] [exit_val_is_increment TRUE|FALSE]
[exit_type value|increment|rate]
[exit_time ?] [poll_interval ?]
[average_factor ?] [queue_priority low|normal|high|last]
[maxrun ?] [nice 0|1]
```

### Arguments

|      |                                                                                                                                                       |
|------|-------------------------------------------------------------------------------------------------------------------------------------------------------|
| tag  | (Optional) String identifying a tag that can be used with the trigger Tcl command extension to support multiple event statements within a Tcl script. |
| name | (Mandatory) The name of the interface being monitored, for example, Ethernet 0/0. Abbreviations and spaces are not allowed.                           |

|           |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |
|-----------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| parameter | <p>(Mandatory) The name of the counter being compared as follows:</p> <ul style="list-style-type: none"> <li>• <code>input_errors</code>--Includes runs, giants, no buffer, CRC, frame, overrun, and ignored counts. Other input-related errors can also cause the input errors count to be increased, and some datagrams may have more than one error; therefore, this sum may not balance with the sum of enumerated input error counts.</li> <li>• <code>input_errors_crc</code>--Cyclic redundancy checksum generated by the originating LAN station or far-end device does not match the checksum calculated from the data received.</li> <li>• <code>input_errors_frame</code>--Number of packets received incorrectly having a CRC error and a noninteger number of octets.</li> <li>• <code>input_errors_overrun</code>--Number of times the receiver hardware was unable to hand received data to a hardware buffer because the input rate exceeded the receiver's ability to handle the data.</li> <li>• <code>input_packets_dropped</code>--Number of packets dropped because of a full input queue.</li> <li>• <code>interface_resets</code>--Number of times that an interface has been completely reset.</li> <li>• <code>output_buffer_failures</code>--Number of failed buffers and number of buffers swapped out.</li> <li>• <code>output_buffer_swappedout</code>--Number of packets swapped to DRAM.</li> </ul> |
|-----------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|

|                        |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |
|------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| parameter (continued)  | <ul style="list-style-type: none"> <li>• output_errors--Sum of all errors that prevented the final transmission of datagrams out of the interface being examined. Note that this may not balance with the sum of the enumerated output errors, because some datagrams may have more than one error, and others may have errors that do not fall into any of the specifically tabulated categories.</li> <li>• output_errors_underrun--Number of times that the transmitter has been running faster than the device can handle.</li> <li>• output_packets_dropped--Number of packets dropped because of a full output queue.</li> <li>• receive_broadcasts--Number of broadcast or multicast packets received by the interface.</li> <li>• receive_giants--Number of packets that are discarded because they exceed the maximum packet size of the medium.</li> <li>• receive_rate_bps--Interface receive rate in bytes per second.</li> <li>• receive_rate_pps--Interface receive rate in packets per second.</li> <li>• receive_runts--Number of packets that are discarded because they are smaller than the minimum packet size of the medium.</li> <li>• receive_throttle--Number of times that the receiver on the port was disabled, possibly because of buffer or processor overload.</li> <li>• reliability--Reliability of the interface as a fraction of 255 (255/255 is 100 percent reliability), calculated as an exponential average over 5 minutes.</li> <li>• rxload--Receive rate of the interface as a fraction of 255 (255/255 is 100 percent).</li> <li>• transmit_rate_bps--Interface transmit rate in bytes per second.</li> <li>• transmit_rate_pps--Interface transmit rate in packets per second.</li> <li>• txload--Transmit rate of the interface as a fraction of 255 (255/255 is 100 percent).</li> </ul> |
| entry_op               | (Mandatory) The comparison operator used to compare the current interface value with the entry value; if true, an event will be raised and event monitoring will be disabled until exit criteria are met.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |
| entry_val              | (Mandatory) The value at which the event will be triggered.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |
| entry_val_is_increment | <p>(Mandatory) If TRUE, the entry_val field is treated as an incremental difference and is compared with the difference between the current counter value and the value when the event was last true (the first polled sample if this is a new event). A negative value checks the incremental difference for a counter that is decreasing. If FALSE, the entry_val field is compared against the current counter value.</p> <p><b>Note</b> This keyword has been deprecated, and if specified, the syntax is converted into equivalent entry-type keyword syntax.</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |

|                       |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |
|-----------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| entry-type            | <p>Specifies a type of operation to be applied to the object ID specified by the entry-val argument.</p> <p>Value is defined as the actual value of the entry-val argument.</p> <p>Increment uses the entry-val field as an incremental difference and the entry-val is compared with the difference between the current counter value and the value when the event was last triggered (or the first polled sample if this is a new event). A negative value checks the incremental difference for a counter that is decreasing.</p> <p>Rate is defined as the average rate of change over a period of time. The time period is the average-factor value multiplied by the poll-interval value. At each poll interval the difference between the current sample and the previous sample is taken and recorded as an absolute value. An average of the previous average-factor value samples is taken to be the rate of change.</p>                                                |
| exit_comb             | (Optional) Used to indicate the combination of exit condition tests required to rearm the event trigger; if the and operator is specified, both exit value and exit time tests must be true to cause rearm; if the or operator is specified, either exit value or exit time tests can be true to cause event monitoring to be rearmed.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |
| exit_op               | (Optional) The comparison operator used to compare the current interface value with the exit value; if true, event monitoring for this event will be reenabled.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |
| exit_val              | (Optional) The value at which the event is rearmed to be monitored again.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |
| exit_val_is_increment | <p>(Optional) If TRUE, the exit_val field is treated as an incremental difference and is compared with the difference between the current counter value and the value when the event was last true. A negative value checks the incremental difference for a counter that is decreasing. If FALSE, the exit_val field is compared against the current counter value.</p> <p><b>Note</b> In Cisco IOS Release 12.4(20)T, this keyword is deprecated, and if specified, the syntax is converted into equivalent exit-type keyword syntax.</p>                                                                                                                                                                                                                                                                                                                                                                                                                                       |
| exit-type             | <p>(Optional) Specifies a type of operation to be applied to the object ID specified by the exit-val argument. If not specified, the value is assumed.</p> <p>Value is defined as the actual value of the exit-val argument.</p> <p>Increment uses the exit-val field as an incremental difference and the exit-val is compared with the difference between the current counter value and the value when the event was last triggered (or the first polled sample if this is a new event). A negative value checks the incremental difference for a counter that is decreasing.</p> <p>Rate is defined as the average rate of change over a period of time. The time period is the average-factor value multiplied by the poll-interval value. At each poll interval the difference between the current sample and the previous sample is taken and recorded as an absolute value. An average of the previous average-factor value samples is taken to be the rate of change.</p> |
| exit_time             | (Optional) The time period at which the event is rearmed to be monitored again (specified in SSSSSSSSS[.MMM] format, where SSSSSSSSS must be an integer representing seconds between 0 and 4294967295, inclusive, and where MMM must be an integer representing milliseconds between 0 and 999).                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |



|                |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
|----------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| poll_interval  | (Optional) The frequency used to collect the samples (specified in SSSSSSSSS[.MMM] format, where SSSSSSSSS must be an integer representing seconds between 60 and 4294967295, inclusive, and where MMM must be an integer representing milliseconds between 0 and 999). The poll interval value must not be less than 1 second. The default is 1 second.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |
| average-factor | (Optional) Number in the range from 1 to 64 used to calculate the period used for rate-based calculations. The average-factor value is multiplied by the poll-interval value to derive the period in milliseconds. The minimum average factor value is 1.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |
| queue_priority | <p>(Optional) Priority level at which the script will be queued:</p> <ul style="list-style-type: none"> <li>• queue_priority low--Specifies that the script is to be queued at the lowest of the three priority levels.</li> <li>• queue_priority normal--Specifies that the script is to be queued at a priority level greater than low priority but less than high priority.</li> <li>• queue_priority high--Specifies that the script is to be queued at the highest of the three priority levels.</li> <li>• queue_priority last--Specifies that the script is to be queued at the lowest priority level.</li> </ul> <p>If more than one script is registered with the "queue_priority_last" argument set, these scripts will execute in the order in which the events are published.</p> <p><b>Note</b> The queue_priority argument specifies the queuing priority, but not the execution priority, of the script being registered.</p> <p>If this argument is not specified, the default queuing priority is normal.</p> |
| maxrun         | (Optional) Maximum run time of the script (specified in SSSSSSSSS[.MMM] format, where SSSSSSSSS must be an integer representing seconds between 0 and 4294967295, inclusive, and where MMM must be an integer representing milliseconds between 0 and 999). If this argument is not specified, the default 20-second run-time limit is used.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |
| nice           | (Optional) Policy run-time priority setting. When the nice argument is set to 1, the policy is run at a run-time priority that is less than the default priority. The default value is 0.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |

**Result String**

None

**Set\_cerrno**

No

**Event\_reqinfo**

```
"event_id %u event_type %u event_type_string {%s} %u event_pub_sec %u event_pub_msec %u"
"event_severity {%s} name {%s} parameter {%s} value %d"
```

| Event Type                                    | Description                                                                                                                                                                                     |
|-----------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>event_id</b>                               | Unique number that indicates the ID for this published event. Multiple policies may be run for the same event, and each policy will have the same event_id.                                     |
| <b>event_type</b>                             | Type of event.                                                                                                                                                                                  |
| <b>event_type_string</b>                      | An ASCII string that represents the name of the event for this event type.                                                                                                                      |
| <b>event_pub_sec</b><br><b>event_pub_msec</b> | The time, in seconds and milliseconds, when the event was published to the EEM.                                                                                                                 |
| <b>event_severity</b>                         | Interface event severity, which can be one of the following values: <ul style="list-style-type: none"> <li>• normal</li> <li>• minor</li> <li>• major</li> </ul>                                |
| <b>name</b>                                   | Name of the interface.                                                                                                                                                                          |
| <b>parameter</b>                              | Name of the parameter.                                                                                                                                                                          |
| <b>value</b>                                  | The incremental/decremental difference compared to the last event triggered or the absolute value of the parameter being monitored, depending on the specified value of entry_val_is_increment. |

## event\_register\_ioswdsysmon

Registers for an IOSWDSysMon event. Use this Tcl command extension to generate an event when a Cisco IOS task exceeds specific CPU utilization or memory thresholds. A Cisco IOS task is called a Cisco IOS process in native Cisco IOS.

### Syntax

```
event_register_ioswdsysmon [tag ?] [timewin ?] [sub12op and|or] [sub1 ?] [sub2 ?]
[queue_priority low|normal|high|last] [maxrun ?] [nice 0|1]
```

### Arguments

|          |                                                                                                                                                                                                                                                                                                                                          |
|----------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| tag      | (Optional) String identifying a tag that can be used with the trigger Tcl command extension to support multiple event statements within a Tcl script.                                                                                                                                                                                    |
| timewin  | (Optional) Defines the time window within which all of the subevents must occur in order for an event to be generated (specified in SSSSSSSSS[.MMM] format, where SSSSSSSSSS must be an integer representing seconds between 0 and 4294967295, inclusive, and where MMM must be an integer representing milliseconds between 0 and 999). |
| sub12_op | (Optional) The combination operator for comparison between subevent 1 and subevent 2.                                                                                                                                                                                                                                                    |

|                |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
|----------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| sub1           | (Optional) The subevent 1 specification.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |
| sub2           | (Optional) The subevent 2 specification.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |
| queue_priority | <p>(Optional) Priority level at which the script will be queued:</p> <ul style="list-style-type: none"> <li>• queue_priority low--Specifies that the script is to be queued at the lowest of the three priority levels.</li> <li>• queue_priority normal--Specifies that the script is to be queued at a priority level greater than low priority but less than high priority.</li> <li>• queue_priority high--Specifies that the script is to be queued at the highest of the three priority levels.</li> <li>• queue_priority last--Specifies that the script is to be queued at the lowest priority level.</li> </ul> <p>If more than one script is registered with the "queue_priority_last" argument set, these scripts will execute in the order in which the events are published.</p> <p><b>Note</b> The queue_priority argument specifies the queuing priority, but not the execution priority, of the script being registered.</p> <p>If this argument is not specified, the default queuing priority is normal.</p> |
| maxrun         | (Optional) Maximum run time of the script (specified in SSSSSSSSS[.MMM] format, where SSSSSSSSS must be an integer representing seconds between 0 and 4294967295, inclusive, and where MMM must be an integer representing milliseconds between 0 and 999). If this argument is not specified, the default 20-second run-time limit is used.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |
| nice           | (Optional) Policy run-time priority setting. When the nice argument is set to 1, the policy is run at a run-time priority that is less than the default priority. The default value is 0.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |

**Subevent Syntax**

```
cpu_proc path ? taskname ? op gt|ge|eq|ne|lt|le val ? [period ?]
mem_proc path ? taskname ? op gt|ge|eq|ne|lt|le val ? [is_percent TRUE|FALSE] [period ?]
```

**Subevent Arguments**

|          |                                                                                                                                                                        |
|----------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| cpu_proc | (Mandatory) Specifies the use of a sample collection of CPU statistics.                                                                                                |
| path     | (Mandatory) Software Modularity images only. The pathname of the POSIX process that contains the Cisco IOS scheduler to be monitored. For example, /sbin/cdp2.iosproc. |
| taskname | (Mandatory) The name of the Cisco IOS task to be monitored.                                                                                                            |
| op       | (Mandatory) The comparison operator used to compare the collected usage sample with the specified value; if true, an event will be raised.                             |
| val      | (Mandatory) The value to be compared.                                                                                                                                  |

|            |                                                                                                                                                                                                                                                                                                                                                                   |
|------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| period     | (Optional) The elapsed time period for the collection samples to be averaged (specified in SSSSSSSSS[.MMM] format, where SSSSSSSSS must be an integer representing seconds between 0 and 4294967295, inclusive, and where MMM must be an integer representing milliseconds between 0 and 999). If this argument is not specified, the most recent sample is used. |
| mem_proc   | (Mandatory) Specifies the use of a sample collection of memory statistics.                                                                                                                                                                                                                                                                                        |
| is_percent | (Optional) Whether the specified value is a percentage.                                                                                                                                                                                                                                                                                                           |

**Result String**

None

**Set\_cerrno**

No

**Event\_reqinfo**

```
"event_id %u event_type %u event_type_string {%s} %u event_pub_sec %u event_pub_msec %u"
"num_subs %u"
```

| Event Type                   | Description                                                                                                                                                 |
|------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------|
| event_id                     | Unique number that indicates the ID for this published event. Multiple policies may be run for the same event, and each policy will have the same event_id. |
| event_type                   | Type of event.                                                                                                                                              |
| event_type_string            | An ASCII string that represents the name of the event for this event type.                                                                                  |
| event_pub_sec event_pub_msec | The time, in seconds and milliseconds, when the event was published to the EEM.                                                                             |
| num_subs                     | Number of subevents.                                                                                                                                        |

Where the subevent info string is for a CPU\_UTIL subevent,

```
"{type %s procname {%s} pid %u taskname {%s} taskid %u value %u sec %ld msec %ld}"
```

| Subevent Type | Description                            |
|---------------|----------------------------------------|
| type          | Type of subevent.                      |
| procname      | POSIX process name for this subevent.  |
| pid           | POSIX process ID for this subevent.    |
| taskname      | Cisco IOS task name for this subevent. |

| Subevent Type | Description                                                |
|---------------|------------------------------------------------------------|
| taskid        | Cisco IOS task ID for this subevent.                       |
| value         | Actual average CPU utilization over the measured interval. |
| sec , msec    | Elapsed time period for this measured interval.            |

Where the subevent info string is for a MEM\_UTIL subevent,

```
"(type %s procname {%s} pid %u taskname {%s} taskid %u is_percent %s value %u diff %d"
"sec %ld msec %ld)"
```

| Subevent Type | Description                                                                                                                                 |
|---------------|---------------------------------------------------------------------------------------------------------------------------------------------|
| type          | Type of subevent.                                                                                                                           |
| procname      | POSIX process name for this subevent.                                                                                                       |
| pid           | POSIX process ID for this subevent.                                                                                                         |
| taskname      | Cisco IOS task name for this subevent.                                                                                                      |
| taskid        | Cisco IOS task ID for this subevent.                                                                                                        |
| is_percent    | TRUE or FALSE depending on whether the value is a percentage value.                                                                         |
| value         | Total memory use in KB or the actual average memory utilization for this measured interval.                                                 |
| diff          | The percentage difference between the oldest sample in the measured interval and the latest sample; a negative value represents a decrease. |
| sec , msec    | Elapsed time period for this measured interval.                                                                                             |

## event\_register\_ipsla

Registers for an event that is triggered by the **event ipsla** command. Use this Tcl command to publish an event when an IPSLA reaction is triggered. The group ID or the operation ID is required to register the event.

### Syntax

```
event_register_ipsla [tag ?] group_name ? operation_id ? [reaction_type ?]
[dest_ip_addr ?][queue_priority low|normal|high|last] [maxrun ?] [nice 0|1]
```

### Arguments

|            |                                                                                                                                                       |
|------------|-------------------------------------------------------------------------------------------------------------------------------------------------------|
| tag        | (Optional) String identifying a tag that can be used with the trigger Tcl command extension to support multiple event statements within a Tcl script. |
| group_name | (Mandatory) Specifies the IP SLAs group name.                                                                                                         |

|                 |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |
|-----------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| operation_id    | (Mandatory) Specifies the IP SLA operation ID. Number must be in the range from 1 to 2147483647.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |
| reaction_type   | <p>(Optional) Specifies the reaction to be taken for the specified IP SLAs operation.</p> <p>Type of IP SLAs reaction--One of the following keywords can be specified: <b>connectionLoss</b>, <b>icpif</b>, <b>jitterAvg</b>, <b>jitterDSAvg</b>, <b>jitterSDAvg</b>, <b>maxOfNegativeDS</b>, <b>maxOfNegativeSD</b>, <b>maxOfPositiveDS</b>, <b>maxOfPositiveSD</b>, <b>mos</b>, <b>packetLateArrival</b>, <b>packetLossDS</b>, <b>packetLossSD</b>, <b>packetMIA</b>, <b>packetOutOfSequence</b>, <b>rtt</b>, <b>timeout</b> or <b>verifyError</b> can be specified.</p> <p>Type of IP SLAs reaction. One of the following keywords can be specified:</p> <ul style="list-style-type: none"> <li>• connectionLoss</li> <li>• icpif</li> <li>• jitterAvg</li> <li>• jitterDSAvg</li> <li>• jitterSDAvg</li> <li>• maxOfNegativeDS</li> <li>• maxOfNegativeSD</li> <li>• maxOfPositiveDS</li> <li>• maxOfPositiveSD</li> <li>• mos</li> <li>• packetLateArrival</li> <li>• packetLossDS</li> <li>• packetLossSD</li> <li>• packetMIA</li> <li>• packetOutOfSequence</li> <li>• rtt</li> <li>• timeout</li> <li>• verifyError</li> </ul> |
| dest_ip_address | (Optional) Specifies the destination IP address of the destination port for which the IP SLAs events are monitored.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |

|                |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
|----------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| queue_priority | <p>(Optional) Priority level at which the script will be queued:</p> <ul style="list-style-type: none"> <li>• queue_priority low--Specifies that the script is to be queued at the lowest of the three priority levels.</li> <li>• queue_priority normal--Specifies that the script is to be queued at a priority level greater than low priority but less than high priority.</li> <li>• queue_priority high--Specifies that the script is to be queued at the highest of the three priority levels.</li> <li>• queue_priority last--Specifies that the script is to be queued at the lowest priority level.</li> </ul> <p>If more than one script is registered with the "queue_priority_last" argument set, these scripts will execute in the order in which the events are published.</p> <p><b>Note</b> The queue_priority argument specifies the queuing priority, but not the execution priority, of the script being registered.</p> <p>If this argument is not specified, the default queuing priority is normal.</p> |
| maxrun         | <p>(Optional) Maximum run time of the script (specified in SSSSSSSSS[.MMM] format, where SSSSSSSSS must be an integer representing seconds between 0 and 31536000, inclusive, and where MMM must be an integer representing milliseconds between 0 and 999). If this argument is not specified, the default 20-second run-time limit is used.</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
| nice           | <p>(Optional) Policy run-time priority setting. When the nice argument is set to 1, the policy is run at a run-time priority that is less than the default priority. The default value is 0.</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |

**Result String**

None

**Set\_cerrno**

No

**Event\_reqinfo**

"event\_ID %u event\_type %u event\_pub\_sec %u event\_pub\_msec %u event\_severity %u" "group\_name %u operation\_id %u condition %u reaction\_type %u dest\_ip\_addr %u" "threshold\_rising %u threshold\_falling %u measured\_threshold\_value %u" "threshold\_count1 %u threshold count2 %u"

| Event Type        | Description                                                                                                                                                 |
|-------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------|
| event_id          | Unique number that indicates the ID for this published event. Multiple policies may be run for the same event, and each policy will have the same event_id. |
| event_type        | The type of event to monitor for the create, update, and delete flow.                                                                                       |
| event_type_string | An ASCII string that represents the name of the event for this event type.                                                                                  |

| Event Type                   | Description                                                                                                                            |
|------------------------------|----------------------------------------------------------------------------------------------------------------------------------------|
| event_pub_sec event_pub_msec | The time, in seconds and milliseconds, when the event was published to the EEM.                                                        |
| event_severity               | The severity of the event.                                                                                                             |
| group_name                   | The name of the IPSLA group.                                                                                                           |
| operation_id                 | The IPSLA operation ID.                                                                                                                |
| condition                    | The condition of IPSLA, which can be one of the following: <ul style="list-style-type: none"> <li>cleared</li> <li>occurred</li> </ul> |
| reaction_type                | The IPSLA reaction type.                                                                                                               |
| dest_ip_address              | The IPSLA destination IP address.                                                                                                      |
| threshold rising             | The IPSLA configured rising threshold value.                                                                                           |
| threshold falling            | The IPSLA configured falling threshold value.                                                                                          |
| measured_threshold_value     | The measured threshold value of the IPSLA operation.                                                                                   |
| threshold_count1             | Corresponds to the argument of the threshold type1.                                                                                    |
| threshold_count2             | Corresponds to the argument of the threshold type2.                                                                                    |

## event\_register\_mat

Registers for a MAT event. Use this Tcl command extension to generate an event when a mac-address is learned in the mac-address-table.

### Syntax

```
event_register_identity [tag ?] interface ?
[mac-address ?]
[type {add | delete}]
[hold-down ?]
[maxrun ?]
```

### Arguments

|             |                                                                                                                                                              |
|-------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------|
| tag         | (Optional) String identifying a tag that can be used with the trigger Tcl command extension to support multiple event statements within a Tcl script.        |
| interface   | A regular expression pattern to match against interface names.                                                                                               |
| mac-address | Mandatory if the interface parameter is not specified. A regular expression pattern that can be used to filter events by mac addresses of the remote device. |



|           |                                                                                                                                                                                                                                                                                                                                              |
|-----------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| type      | (Optional) Filter based on a mac-address-table event type of add or delete. If not specified, the event type is not used in determining whether the event should be triggered.                                                                                                                                                               |
| hold-down | (Optional) When a mac-address-table event comes in, the hold-down timer can be set to make the event to wait between 1 and 4294967295 seconds before processing the policy. If not set then the policy is not delayed in being processed.                                                                                                    |
| maxrun    | (Optional) Maximum run time of the script (specified in SSSSSSSSSS[.MMM] format, where SSSSSSSSSS must be an integer representing seconds between 0 and 31536000, inclusive, and where MMM must be an integer representing milliseconds between 0 and 999). If this argument is not specified, the default 20-second run-time limit is used. |

**Result String**

None

**Set\_cerrno**

No

**Event\_reqinfo For EEM\_EVENT\_MAT**

```
"event_id %u event_type %u event_type_string {%s} event_pub_sec %u event_pub_msec %u
event_severity %u notification %u intf_name %u mac_address {%s}"
```

| Event Type                          | Description                                                                                                                                                 |
|-------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>event_id</b>                     | Unique number that indicates the ID for this published event. Multiple policies may be run for the same event, and each policy will have the same event_id. |
| <b>event_type</b>                   | Type of event.                                                                                                                                              |
| <b>event_type_string</b>            | An ASCII string that represents the name of the event for this event type.                                                                                  |
| <b>event_pub_sec event_pub_msec</b> | The time, in seconds and milliseconds, at which the event was published to the EEM.                                                                         |
| <b>event_severity</b>               | The severity of the event.                                                                                                                                  |
| <b>notification</b>                 | Notification type--add or delete.                                                                                                                           |
| <b>intf_name</b>                    | The interface name for the address table entry.                                                                                                             |
| <b>mac_address</b>                  | The mac-address for the address table entry.                                                                                                                |

# event\_register\_neighbor\_discovery

Registers for a neighbor discover event. Use this Tcl command extension to generate an event when a Cisco Discovery Protocol (CDP) or Link Layer Discovery Protocol (LLDP) cache entry or a interface link status changes.

## Syntax

```
event_register_neighbor_discovery [tag ?] interface ?
[cdp {add | update | delete | all}]
[lldp {add | update | delete | all}]
[link-event]
[line-event]
[queue_priority {normal | low | high | last}]
[maxrun ?] [nice {0 | 1}]
```

## Arguments

|            |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |
|------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| tag        | (Optional) String identifying a tag that can be used with the trigger Tcl command extension to support multiple event statements within a Tcl script.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
| interface  | A regular expression pattern to match against interface names.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |
| cdp        | Trigger an event when a matching CDP event occurs. One of the following options should be specified. <ul style="list-style-type: none"> <li>• add--Trigger events only when a new CDP cache entry is created in the CDP table.</li> <li>• all--Trigger an event when a CDP cache entry is added or deleted from the CDP cache table and when a remote CDP device sends a keepalive to update the CDP cache entry.</li> <li>• delete--trigger events only when a CDP cache entry is deleted from the CDP table.</li> <li>• update--trigger an event when a CDP cache entry is added to the CDP table or when the remote CDP device sends a CDP keepalive to update the CDP cache entry.</li> </ul>  |
| lldp       | Trigger an event when a matching lldp event occurs. One of the following options should be specified. <ul style="list-style-type: none"> <li>• add--Trigger events only when a new cdp cache entry is created in the cdp table.</li> <li>• all--Trigger an event when a cdp cache entry is added or deleted from the cdp cache table and when a remote cdp device sends a keepalive to update the cdp cache entry.</li> <li>• delete--trigger events only when a cdp cache entry is deleted from the cdp table.</li> <li>• update--trigger an event when a cdp cache entry is added to the cdp table or when the remote cdp device sends a cdp keepalive to update the cdp cache entry.</li> </ul> |
| line-event | Trigger an event when the interface line protocol status changes.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |
| link-event | Trigger an event when the interface link status changes.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |

|                |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |
|----------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| queue_priority | <p>(Optional) Priority level at which the script will be queued:</p> <ul style="list-style-type: none"> <li>• queue_priority low--Specifies that the script is to be queued at the lowest of the three priority levels.</li> <li>• queue_priority normal--Specifies that the script is to be queued at a priority level greater than low priority but less than high priority.</li> <li>• queue_priority high--Specifies that the script is to be queued at the highest of the three priority levels.</li> <li>• queue_priority last--Specifies that the script is to be queued at the lowest priority level.</li> </ul> <p>If more than one script is registered with the "queue_priority_last" argument set, these scripts will execute in the order in which the events are published.</p> <p>The queue_priority argument specifies the queuing priority, but not the execution priority, of the script being registered.</p> <p>If this argument is not specified, the default queuing priority is normal.</p> |
| maxrun         | <p>(Optional) Maximum run time of the script (specified in SSSSSSSSS[.MMM] format, where SSSSSSSSS must be an integer representing seconds between 0 and 31536000, inclusive, and where MMM must be an integer representing milliseconds between 0 and 999). If this argument is not specified, the default 20-second run-time limit is used.</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |
| nice           | <p>(Optional) Policy run-time priority setting. When the nice argument is set to 1, the policy is run at a run-time priority that is less than the default priority. The default value is 0.</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |

**Result String**

None

**Set\_cerrno**

No

**Event\_reqinfo For EEM\_EVENT\_NEIGHBOR\_DISCOVERY**

```
"event_id %u event_type %u event_type_string {%s} event_pub_sec %u event_pub_msec %u event_severity %u nd_notification {%s}"
```

| Event Type                   | Description                                                                                                                                                 |
|------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------|
| event_id                     | Unique number that indicates the ID for this published event. Multiple policies may be run for the same event, and each policy will have the same event_id. |
| event_type                   | Type of event.                                                                                                                                              |
| event_type_string            | An ASCII string that represents the name of the event for this event type.                                                                                  |
| event_pub_sec event_pub_msec | The time, in seconds and milliseconds, at which the event was published to the EEM.                                                                         |

| Event Type                            | Description                                                                                                                                                                                                                                                  |
|---------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>event_severity</b>                 | The severity of the event.                                                                                                                                                                                                                                   |
| Common Event_Reqinfo                  |                                                                                                                                                                                                                                                              |
| <b>nd_notification</b>                | The type of notification--cdp-add, cdp-update, cdp-delete, lldp-add, lldp-update, lldp-delete, link, line.                                                                                                                                                   |
| <b>nd_intf_linkstatus</b>             | The current interface link status, up or down.                                                                                                                                                                                                               |
| <b>nd_intf_linestatus</b>             | The current interface line status, down, goingdown, init, testing, up, reset, admindown, deleted.                                                                                                                                                            |
| <b>nd_local_intf_name</b>             | The local interface name for the event.                                                                                                                                                                                                                      |
| <b>nd_short_local_intf_name</b>       | The short name of the local interface for the event.                                                                                                                                                                                                         |
| <b>nd_port_id</b>                     | The port id as identified by either the cdp or lldp protocol. This is not set for link or line protocol events.                                                                                                                                              |
| CDP-specific Event_reqinfo            |                                                                                                                                                                                                                                                              |
| <b>nd_protocol</b>                    | Identifies which protocol triggered the event, for CDP it will always be set to cdp.                                                                                                                                                                         |
| <b>nd_proto_notif</b>                 | Identifies which type of protocol event triggered the event, add, update or delete.                                                                                                                                                                          |
| <b>nd_proto_new_entry</b>             | If set to 1, the event was triggered because the cache entry is new, otherwise it will be set to 0.                                                                                                                                                          |
| <b>nd_cdp_entry_name</b>              | The name of the cdp cache entry in the cdp table.                                                                                                                                                                                                            |
| <b>nd_cdp_hold_time</b>               | The time remaining until the cdp cache entry expires and is deleted from the cdp table. This time will be reset to some maximum by an update from the cdp neighbor. It is usually set to 0 for new entries.                                                  |
| <b>nd_cdp_mgmt_domain</b>             | The CDP VTP management domain.                                                                                                                                                                                                                               |
| <b>nd_cdp_platform</b>                | The platform name reported by the remote device.                                                                                                                                                                                                             |
| <b>nd_cdp_version</b>                 | The version of code running on the remote device.                                                                                                                                                                                                            |
| <b>nd_cdp_capabilities_string</b>     | The contents of the CDP capabilities field in a string format: Router, Trans-Bridge, Source-Route-Bridge, Switch, Host, IGMP, Repeater, Phone, Remotely-Managed device, CVTA phone port, Two-port Mac Relay or any combination of these separated by commas. |
| <b>nd_cdp_capabilities_bits</b>       | The CDP capabilities bits in a hexadecimal number preceded with 0x.                                                                                                                                                                                          |
| <b>nd_cdp_capabilities_bit_[0-31]</b> | A series of values that will be set to YES if that bit in the capabilities field is set or NO if it is not set.                                                                                                                                              |

| Event Type                                 | Description                                                                                                                                                                         |
|--------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>LLDP-specific Event_reqinfo</b>         |                                                                                                                                                                                     |
| <b>nd_protocol</b>                         | Identifies which protocol triggered the event, for LLDP it will always be set to lldp.                                                                                              |
| <b>nd_proto_notif</b>                      | Identifies which type of protocol event triggered the event, add, update or delete.                                                                                                 |
| <b>nd_proto_new_entry</b>                  | If set to 1, the event was triggered because the cache entry is new, otherwise it will be set to 0.                                                                                 |
| <b>nd_lldp_chassis_id</b>                  | The chassis id field from the LLDP cache entry.                                                                                                                                     |
| <b>nd_lldp_system_name</b>                 | The system name from the LLDP cache entry.                                                                                                                                          |
| <b>nd_lldp_system_description</b>          | The system description field from the LLDP cache entry.                                                                                                                             |
| <b>nd_lldp_ttl</b>                         | The LLDP time to live field from the LLDP cache entry.                                                                                                                              |
| <b>nd_lldp_port_description</b>            | The port description field from the LLDP cache entry.                                                                                                                               |
| <b>nd_lldp_system_capabilities_string</b>  | The LLDP system capabilities field from the LLDP cache entry. Provided as a string that can contain O, P, B, W, R, T, C, S or any combination of these separated by commas.         |
| <b>nd_lldp_enabled_capabilities_string</b> | The LLDP enabled system capabilities field from the LLDP cache entry. Provided as a string that can contain O, P, B, W, R, T, C, S or any combination of these separated by commas. |
| <b>nd_lldp_system_capabilities_bits</b>    | The LLDP system capabilities bits field from the LLDP cache entry. Provided as a hexadecimal number preceded by 0x.                                                                 |
| <b>nd_lldp_enabled_capabilities_bits</b>   | The LLDP enabled capabilities bits field from the LLDP cache entry. Provided as a hexadecimal number preceded by 0x.                                                                |
| <b>nd_lldp_capabilities_bits</b>           | The LLDP capabilities bits field from the LLDP cache entry. Provided as a hexadecimal number preceded by 0x.                                                                        |
| <b>nd_lldp_capabilities_bit_[0-31]</b>     | A series of values that will be set to YES if that bit in the capabilities field is set or NO if it is not set.                                                                     |

## event\_register\_nf

Registers for an event when a NetFlow event is triggered by the **event nf** command. Use this Tel command to publish an event when an NetFlow reaction is triggered..

### Syntax

```
event_register_nf [tag ?] monitor_name ? event_type create|update|delete
exit_event_type create|update|delete event1-event4 ? [maxrun ?] [nice 0|1]
```

## Arguments

|                 |                                                                                                                                                                                                                                                                                                                                              |
|-----------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| tag             | (Optional) String identifying a tag that can be used with the trigger Tcl command extension to support multiple event statements within a Tcl script.                                                                                                                                                                                        |
| monitor_name    | (Mandatory) The name of the NetFlow monitor.                                                                                                                                                                                                                                                                                                 |
| event_type      | (Mandatory) The type of event to monitor for the create, update, and delete flow.                                                                                                                                                                                                                                                            |
| exit_event_type | (Mandatory) The event-type (create, delete, update) at which the event is rearmed to be monitored again.                                                                                                                                                                                                                                     |
| event1- event4  | (Mandatory) Specifies the event and its attributes to monitor. Valid values are <b>event1</b> , <b>event2</b> , <b>event3</b> , and <b>event4</b> .<br><br>The subevent keywords can be used alone, together, or in any combination with each other, but each keyword can be used only once.                                                 |
| maxrun          | (Optional) Maximum run time of the script (specified in SSSSSSSSS[.MMM] format, where SSSSSSSSS must be an integer representing seconds between 0 and 4294967295, inclusive, and where MMM must be an integer representing milliseconds between 0 and 999). If this argument is not specified, the default 20-second run-time limit is used. |
| nice            | (Optional) Policy run-time priority setting. When the nice argument is set to 1, the policy is run at a run-time priority that is less than the default priority. The default value is 0.                                                                                                                                                    |

## Subevent Syntax

```
field ? rate_interval ? event1 only entry_value ? entry_op eq|ge|gt|le|lt|wc
[exit_value ?] [exit_op eq|ge|gt|le|lt|wc] [exit_rate_interval ? event1 only]
```

## Subevent Arguments

|               |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
|---------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| field         | (Mandatory) Specifies the cache or field attribute to be monitored. One of the following attributes can be specified: <ul style="list-style-type: none"> <li>• <b>counter</b> {<b>bytes</b>   <b>packets</b>}--Specifies the counter fields.</li> <li>• <b>datalink</b> {<b>dot1q</b>   <b>mac</b>}--Specifies the datalink (layer2) fields.</li> <li>• <b>flow</b> {<b>direction</b>   <b>sampler</b>}--Specifies the flow identifying fields.</li> <li>• <b>interface</b> {<b>input</b>   <b>output</b>}--Specifies the interface fields.</li> <li>• <b>ipv4</b> <i>field-type</i>-- Specifies the IPv4 fields.</li> <li>• <b>ipv6</b> <i>field-type</i>-- IPv6 fields</li> <li>• <b>routing</b> <i>routing-attribute</i> -- Specifies the routing attributes.</li> <li>• <b>timestamp</b> <b>sysuptime</b> {<b>first</b>   <b>last</b>}--Specifies the timestamp fields.</li> <li>• <b>transport</b> <i>field-type</i>-- Specifies the Transport layer fields.</li> </ul> |
| rate_interval | (Mandatory) Specifies the rate interval value in seconds used to calculate the rate. This field is only valid for event1.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |

|                    |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
|--------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| entry_value        | (Mandatory) Specifies the field or rate value.                                                                                                                                                                                                                                                                                                                                                                                                                                               |
| entry_op           | (Mandatory) Specifies the field operator.<br>The comparison operator valid values are: <ul style="list-style-type: none"> <li>• <b>eq</b> - Equal to</li> <li>• <b>ge</b> - Greater than or equal to</li> <li>• <b>gt</b> - Greater than</li> <li>• <b>le</b> - Less than or equal to</li> <li>• <b>lt</b> - Less than</li> <li>• <b>wc</b> - Wildcard</li> </ul>                                                                                                                            |
| exit_value         | (Optional) The value at which the event is rearmed to be monitored again.                                                                                                                                                                                                                                                                                                                                                                                                                    |
| exit_op            | (Optional) The comparison operator used to compare the current event field or rate value with the exit value; if true, event monitoring for this event is reenabled.<br>The comparison operator valid values are: <ul style="list-style-type: none"> <li>• <b>eq</b> - Equal to</li> <li>• <b>ge</b> - Greater than or equal to</li> <li>• <b>gt</b> - Greater than</li> <li>• <b>le</b> - Less than or equal to</li> <li>• <b>lt</b> - Less than</li> <li>• <b>wc</b> - Wildcard</li> </ul> |
| exit_rate_interval | (Optional) Specifies the exit rate interval value in seconds used to calculate the exit rate value. This field is only valid for event1.                                                                                                                                                                                                                                                                                                                                                     |

**Result String**

None

**Set\_cerrno**

No

**Event\_reqinfo**

"event\_ID %u event\_type %u event\_type\_string {%s} event\_pub\_sec %u event\_pub\_msec %u event\_severity %u monitor\_name %u event1-event4\_field %u event1-event4\_value

| Event Type | Description |
|------------|-------------|
|------------|-------------|

|                                     |                                                                                                                                                             |
|-------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>event_id</b>                     | Unique number that indicates the ID for this published event. Multiple policies may be run for the same event, and each policy will have the same event_id. |
| <b>event_type</b>                   | The type of event to monitor for the create, update, and delete flow.                                                                                       |
| <b>event_type_string</b>            | An ASCII string that represents the name of the event for this event type.                                                                                  |
| <b>event_pub_sec event_pub_msec</b> | The time, in seconds and milliseconds, when the event was published to the EEM.                                                                             |
| <b>event_severity</b>               | The severity of the NetFlow event.                                                                                                                          |
| <b>monitor_name</b>                 | The name of the NetFlow monitor.                                                                                                                            |
| <b>event1-event4_field</b>          | Specifies the event and its attributes to monitor. Valid values are <b>event1</b> , <b>event2</b> , <b>event3</b> , and <b>event4</b> .                     |
| <b>event1-event4_value</b>          | Specifies the event value and its attributes to monitor. Valid values are <b>event1</b> , <b>event2</b> , <b>event3</b> , and <b>event4</b> .               |

## event\_register\_none

Registers for an event that is triggered by the **event manager run** command. These events are handled by the None event detector that screens for this event.

### Syntax

```
event_register_none [tag ?] [sync {yes|no}] [default ?] [queue_priority low|normal|high|last]
[maxrun ?] [nice 0|1]
```

### Arguments

|         |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |
|---------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| tag     | (Optional) String identifying a tag that can be used with the trigger Tcl command extension to support multiple event statements within a Tcl script.                                                                                                                                                                                                                                                                                                                                                                                     |
| sync    | (Optional) A "yes" or a "no" is required to complete this keyword. <ul style="list-style-type: none"> <li>• If the yes keyword is specified, the policy will run synchronously with the CLI command.</li> <li>• If the no keyword is specified, the policy will run asynchronously with the CLI command.</li> </ul>                                                                                                                                                                                                                       |
| default | (Optional) The time period during which the CLI event detector waits for the policy to exit (specified in SSSSSSSSS[.MMM] format, where SSSSSSSSS must be an integer representing seconds between 0 and 4294967295, inclusive, and where MMM must be an integer representing milliseconds between 0 and 999). If the default time period expires before the policy exits, the default action will be executed. The default action is to run the command. If this argument is not specified, the default time period is set to 30 seconds. |



|                |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
|----------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| queue_priority | <p>(Optional) Priority level at which the script will be queued:</p> <ul style="list-style-type: none"> <li>• queue_priority low--Specifies that the script is to be queued at the lowest of the three priority levels.</li> <li>• queue_priority normal--Specifies that the script is to be queued at a priority level greater than low priority but less than high priority.</li> <li>• queue_priority high--Specifies that the script is to be queued at the highest of the three priority levels.</li> <li>• queue_priority last--Specifies that the script is to be queued at the lowest priority level.</li> </ul> <p>If more than one script is registered with the "queue_priority_last" argument set, these scripts will execute in the order in which the events are published.</p> <p><b>Note</b> The queue_priority argument specifies the queuing priority, but not the execution priority, of the script being registered.</p> <p>If this argument is not specified, the default queuing priority is normal.</p> |
| maxrun         | <p>(Optional) Maximum run time of the script (specified in SSSSSSSSS[.MMM] format, where SSSSSSSSS must be an integer representing seconds between 0 and 4294967295, inclusive, and where MMM must be an integer representing milliseconds between 0 and 999). If this argument is not specified, the default 20-second run-time limit is used.</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |
| nice           | <p>(Optional) Policy run-time priority setting. When the nice argument is set to 1, the policy is run at a run-time priority that is less than the default priority. The default value is 0.</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |

**Result String**

None

**Set\_cerrno**

No

**Event\_reqinfo**

```
"event_id %u event_type %u event_type_string {%s} event_pub_sec %u event_pub_msec %u
event_severity %u arg %u"
```

| Event Type               | Description                                                                                                                                                 |
|--------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>event_id</b>          | Unique number that indicates the ID for this published event. Multiple policies may be run for the same event, and each policy will have the same event_id. |
| <b>event_type</b>        | Type of event.                                                                                                                                              |
| <b>event_type_string</b> | An ASCII string that represents the name of the event for this event type.                                                                                  |

|                                                                                                                                                                                                                                     |                                                                                 |
|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------|
| <b>event_pub_sec</b> <b>event_pub_msec</b>                                                                                                                                                                                          | The time, in seconds and milliseconds, when the event was published to the EEM. |
| <b>event_severity</b>                                                                                                                                                                                                               | The severity of the event.                                                      |
| <b>argc</b><br><b>arg1</b><br><b>arg2</b><br><b>arg3</b><br><b>arg4</b><br><b>arg6</b><br><b>arg7</b><br><b>arg8</b><br><b>arg9</b><br><b>arg10</b><br><b>arg11</b><br><b>arg12</b><br><b>arg13</b><br><b>arg14</b><br><b>arg15</b> | The parameters that are passed from the XML SOAP command to the script.         |

## event\_register\_oir

Registers for an online insertion and removal (OIR) event. Use this Tcl command extension to run a policy on the basis of an event raised when a hardware card OIR occurs. These events are handled by the OIR event detector that screens for this event.

### Syntax

```
event_register_oir [tag ?] [queue_priority low|normal|high|last] [maxrun ?] [nice 0|1]
```

### Arguments

|     |                                                                                                                                                       |
|-----|-------------------------------------------------------------------------------------------------------------------------------------------------------|
| tag | (Optional) String identifying a tag that can be used with the trigger Tcl command extension to support multiple event statements within a Tcl script. |
|-----|-------------------------------------------------------------------------------------------------------------------------------------------------------|

|                |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
|----------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| queue_priority | <p>(Optional) Priority level at which the script will be queued:</p> <ul style="list-style-type: none"> <li>• queue_priority low--Specifies that the script is to be queued at the lowest of the three priority levels.</li> <li>• queue_priority normal--Specifies that the script is to be queued at a priority level greater than low priority but less than high priority.</li> <li>• queue_priority high--Specifies that the script is to be queued at the highest of the three priority levels.</li> <li>• queue_priority last--Specifies that the script is to be queued at the lowest priority level.</li> </ul> <p>If more than one script is registered with the "queue_priority_last" argument set, these scripts will execute in the order in which the events are published.</p> <p><b>Note</b> The queue_priority argument specifies the queuing priority, but not the execution priority, of the script being registered.</p> <p>If this argument is not specified, the default queuing priority is normal.</p> |
| maxrun         | <p>(Optional) Maximum run time of the script (specified in SSSSSSSSS[.MMM] format, where SSSSSSSSS must be an integer representing seconds between 0 and 4294967295, inclusive, and where MMM must be an integer representing milliseconds between 0 and 999). If this argument is not specified, the default 20-second run-time limit is used.</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |
| nice           | <p>(Optional) Policy run-time priority setting. When the nice argument is set to 1, the policy is run at a run-time priority that is less than the default priority. The default value is 0.</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |

**Result String**

None

**Set\_cerrno**

No

**Event\_reqinfo**

"event\_id %u event\_type %u event\_type\_string {%s} event\_pub\_sec %u event\_pub\_msec %u"  
 "slot %u event %s"

| Event Type                   | Description                                                                                                                                                 |
|------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------|
| event_id                     | Unique number that indicates the ID for this published event. Multiple policies may be run for the same event, and each policy will have the same event ID. |
| event_type                   | Type of event.                                                                                                                                              |
| event_type_string            | An ASCII string that represents the name of the event for this event type.                                                                                  |
| event_pub_sec event_pub_msec | The time, in seconds and milliseconds, when the event was published to the EEM.                                                                             |

| Event Type | Description                                                                                                   |
|------------|---------------------------------------------------------------------------------------------------------------|
| slot       | Slot number for the affected card.                                                                            |
| event      | Indicates a string, removed or online, that represents either an OIR removal event or an OIR insertion event. |

## event\_register\_process

Registers for a process event. Use this Tcl command extension to run a policy on the basis of an event raised when a Cisco IOS Software Modularity process starts or stops. These events are handled by the System Manager event detector that screens for this event. This Tcl command extension is supported only in Software Modularity images.

### Syntax

```
event_register_process [tag ?] abort|term|start|user_restart|user_shutdown
[sub_system ?] [version ?] [instance ?] [path ?] [node ?]
[queue_priority low|normal|high|last] [maxrun ?] [nice 0|1]
```

### Arguments

|               |                                                                                                                                                                                                                                           |
|---------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| tag           | (Optional) String identifying a tag that can be used with the trigger Tcl command extension to support multiple event statements within a Tcl script.                                                                                     |
| abort         | (Mandatory) Abnormal process termination. Process may terminate because of exiting with a nonzero exit status, receiving a kernel-generated signal, or receiving a SIGTERM or SIGKILL signal that is not sent because of user request.    |
| term          | (Mandatory) Normal process termination.                                                                                                                                                                                                   |
| start         | (Mandatory) Process start.                                                                                                                                                                                                                |
| user_restart  | (Mandatory) Process termination due to the process restart request from the CLI command.                                                                                                                                                  |
| user_shutdown | (Mandatory) Process termination due to the process kill request from the CLI command.                                                                                                                                                     |
| sub_system    | (Optional) Number assigned to the EEM policy that published the process event. Number is set to 798 because all other numbers are reserved for Cisco use.                                                                                 |
| version       | (Optional) Version number of the process assigned by the version manager. Must be of the form major_number.minor_number.level. If specified, each component of the version number must be an integer between 1 and 4294967295, inclusive. |
| instance      | (Optional) Process instance ID. If specified, this argument must be an integer between 1 and 4294967295, inclusive.                                                                                                                       |
| path          | (Optional) Process pathname (a regular expression string). If the value of the process-name argument contains embedded blanks, enclose it in double quotation marks. Use path ".*" to match all processes.                                |

|                |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
|----------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| node           | <p>(Optional) The node name is a string that consists of the word "node" followed by two fields separated by a slash character using the following format:</p> <p>node&lt;slot-number&gt;/&lt;cpu-number&gt;</p> <p>The slot-number is the hardware slot number. The cpu-number is the hardware CPU number. For example, the SP CPU in a Supervisor card on a Cisco Catalyst 6500 series switch located in slot 0 would be specified as node0/0. The RP CPU in a Supervisor card on a Cisco Catalyst 6500 series switch located in slot 0 would be addressed as node0/1. If the node argument is not specified, the default node specification is always the regular expression pattern match of * representing all applicable nodes.</p>                                                                                                                                                                                                                                                                                      |
| queue_priority | <p>(Optional) Priority level at which the script will be queued:</p> <ul style="list-style-type: none"> <li>• queue_priority low--Specifies that the script is to be queued at the lowest of the three priority levels.</li> <li>• queue_priority normal--Specifies that the script is to be queued at a priority level greater than low priority but less than high priority.</li> <li>• queue_priority high--Specifies that the script is to be queued at the highest of the three priority levels.</li> <li>• queue_priority last--Specifies that the script is to be queued at the lowest priority level.</li> </ul> <p>If more than one script is registered with the "queue_priority_last" argument set, these scripts will execute in the order in which the events are published.</p> <p><b>Note</b> The queue_priority argument specifies the queuing priority, but not the execution priority, of the script being registered.</p> <p>If this argument is not specified, the default queuing priority is normal.</p> |
| maxrun         | <p>(Optional) Maximum run time of the script (specified in SSSSSSSSS[.MMM] format, where SSSSSSSSS must be an integer representing seconds between 0 and 4294967295, inclusive, and where MMM must be an integer representing milliseconds between 0 and 999). If this argument is not specified, the default 20-second run-time limit is used.</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |
| nice           | <p>(Optional) Policy run-time priority setting. When the nice argument is set to 1, the policy is run at a run-time priority that is less than the default priority. The default value is 0.</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |

If an optional argument is not specified, the event matches all possible values of the argument. If multiple arguments are specified, the process event will be raised when all the conditions are matched.

**Result String**

None

**Set\_cerrno**

No

**Event\_reqinfo**

"event\_id %u event\_type %u event\_type\_string {%s} event\_pub\_sec %u event\_pub\_msec %u"

```
"sub_system 0x%x instance %u process_name {%s} path {%s} exit_status 0x%x"
"respawn_count %u last_respawn_sec %ld last_respawn_msec %ld fail_count %u"
"dump_count %u node_name {%s}"
```

| Event Type                                    | Description                                                                                                                                                                                                                                                                                                                             |
|-----------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>event_id</b>                               | Unique number that indicates the ID for this published event. Multiple policies may be run for the same event, and each policy will have the same event_id.                                                                                                                                                                             |
| <b>event_type</b>                             | Type of event.                                                                                                                                                                                                                                                                                                                          |
| <b>event_type_string</b>                      | An ASCII string that represents the name of the event for this event type.                                                                                                                                                                                                                                                              |
| <b>event_pub_sec event_pub_msec</b>           | The time, in seconds and milliseconds, when the event was published to the EEM.                                                                                                                                                                                                                                                         |
| <b>sub_system</b>                             | Number assigned to the EEM policy that published the application-specific event. Number is set to 798 because all other numbers are reserved for Cisco use.                                                                                                                                                                             |
| <b>instance</b>                               | Process instance ID.                                                                                                                                                                                                                                                                                                                    |
| <b>process_name</b>                           | Process name.                                                                                                                                                                                                                                                                                                                           |
| <b>path</b>                                   | Process absolute name including path.                                                                                                                                                                                                                                                                                                   |
| <b>exit_status</b>                            | Process last exit status.                                                                                                                                                                                                                                                                                                               |
| <b>respawn_count</b>                          | Number of times that the process was restarted.                                                                                                                                                                                                                                                                                         |
| <b>last_respawn_sec<br/>last_respawn_msec</b> | The calendar time when the last restart occurred.                                                                                                                                                                                                                                                                                       |
| <b>fail_count</b>                             | Number of restart attempts of the process that failed. This count will be reset to 0 when the process is successfully restarted.                                                                                                                                                                                                        |
| <b>dump_count</b>                             | Number of core dumps taken of the process.                                                                                                                                                                                                                                                                                              |
| <b>node_name</b>                              | Name of the node that the process is on. The node name is a string that consists of the word "node" followed by two fields separated by a slash character using the following format:<br><br><b>node</b> <i>slot-number / cpu-number</i><br><br>The slot-number is the hardware slot number. The cpu-number is the hardware CPU number. |

## event\_register\_resource

Registers for an Embedded Resource Manager (ERM) event. Use this Tcl command extension to run a policy on the basis of an ERM event report for a specified policy. ERM events are screened by the EEM Resource event detector, allowing an EEM policy to be run when a match occurs for the specified ERM policy.

### Syntax

```
event_register_resource policy policy-name [queue_priority low|normal|high|last]
[maxrun ?] [nice 0|1]
```

### Arguments

|                |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
|----------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| policy         | (Mandatory) Specifies the use of a policy.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |
| policy-name    | (Mandatory) Name of an ERM policy.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |
| queue_priority | <p>(Optional) Priority level at which the script will be queued:</p> <ul style="list-style-type: none"> <li>• queue_priority low--Specifies that the script is to be queued at the lowest of the three priority levels.</li> <li>• queue_priority normal--Specifies that the script is to be queued at a priority level greater than low priority but less than high priority.</li> <li>• queue_priority high--Specifies that the script is to be queued at the highest of the three priority levels.</li> <li>• queue_priority last--Specifies that the script is to be queued at the lowest priority level.</li> </ul> <p>If more than one script is registered with the "queue_priority_last" argument set, these scripts will execute in the order in which the events are published.</p> <p><b>Note</b> The queue_priority argument specifies the queuing priority, but not the execution priority, of the script being registered.</p> <p>If this argument is not specified, the default queuing priority is normal.</p> |
| maxrun         | (Optional) Maximum run time of the script (specified in SSSSSSSSS[.MMM] format, where SSSSSSSSS must be an integer representing seconds between 0 and 4294967295, inclusive, and where MMM must be an integer representing milliseconds between 0 and 999). If this argument is not specified, the default 20-second run-time limit is used.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |
| nice           | (Optional) Policy run-time priority setting. When the nice argument is set to 1, the policy is run at a run-time priority that is less than the default priority. The default value is 0.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |

### Result String

None

### Set\_cerrno

No

### Event\_reqinfo

```
"event_id %u event_type %u event_type_string {%s} %u event_pub_sec %u event_pub_msec %u"
"owner_id %lld user_id %lld" time_sent %llu dampen_time %d notify_data_flags %u"
"level {%s} direction {%s} configured_threshold %u current_value %u"
"policy_violation_flag {%s} policy_id %d"
```

| Event Type                          | Description                                                                                                                                                 |
|-------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>event_id</b>                     | Unique number that indicates the ID for this published event. Multiple policies may be run for the same event, and each policy will have the same event_id. |
| <b>event_type</b>                   | Type of event.                                                                                                                                              |
| <b>event_type_string</b>            | An ASCII string that represents the name of the event for this event type.                                                                                  |
| <b>event_pub_sec event_pub_msec</b> | The time, in seconds and milliseconds, when the event was published to the EEM.                                                                             |
| <b>owner_id</b>                     | The Embedded Resource Manager (ERM) owner ID.                                                                                                               |
| <b>user_id</b>                      | The ERM user ID.                                                                                                                                            |
| <b>time_sent</b>                    | The ERM event time, in nanoseconds.                                                                                                                         |
| <b>dampen_time</b>                  | The ERM dampen time, in nanoseconds.                                                                                                                        |
| <b>notify_data_flags</b>            | The ERM notify data flag.                                                                                                                                   |
| <b>level</b>                        | The ERM event level. The four event levels are normal, minor, major, and critical.                                                                          |
| <b>direction</b>                    | The ERM event direction. The event direction can be one of the following: up, down, or no change.                                                           |
| <b>configured_threshold</b>         | The configured ERM threshold.                                                                                                                               |
| <b>current_value</b>                | The current value reported by ERM.                                                                                                                          |
| <b>policy_violation_flag</b>        | The ERM policy violation flag; either false or true.                                                                                                        |
| <b>policy_id</b>                    | The ERM policy ID.                                                                                                                                          |

## event\_register\_rf

Registers for a Redundancy Facility (RF) event. Use this Tcl command extension to run a policy when an RF progression or status event notification occurs.

### Syntax

```
event_register_rf [tag ?] event ?
[queue_priority low|normal|high|last]
[maxrun ?] [nice 0|1]
```



**Arguments**

|       |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |
|-------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| tag   | (Optional) String identifying a tag that can be used with the trigger Tcl command extension to support multiple event statements within a Tcl script.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |
| event | <p>(Mandatory) Name of the RF progression or status event. Valid values are:</p> <ul style="list-style-type: none"> <li>• RF_PROG_ACTIVE</li> <li>• RF_PROG_ACTIVE_DRAIN</li> <li>• RF_PROG_ACTIVE_FAST = 200</li> <li>• RF_PROG_ACTIVE_PRECONFIG</li> <li>• RF_PROG_ACTIVE_POSTCONFIG</li> <li>• RF_PROG_EXTRALOAD</li> <li>• RF_PROG_HANDBACK</li> <li>• RF_PROG_INITIALIZATION</li> <li>• RF_PROG_PLATFORM_SYNC</li> <li>• RF_PROG_STANDBY_BULK</li> <li>• RF_PROG_STANDBY_COLD</li> <li>• RF_PROG_STANDBY_CONFIG</li> <li>• RF_PROG_STANDBY_FILESYS</li> <li>• RF_PROG_STANDBY_HOT</li> <li>• RF_PROG_STANDBY_OIR_SYNC_DONE</li> <li>• RF_REGISTRATION_STATUS</li> <li>• RF_STATUS_MAINTENANCE_ENABLE</li> <li>• RF_STATUS_MANUAL_SWACT</li> <li>• RF_STATUS_OPER_REDUNDANCY_MODE_CHANGE</li> <li>• RF_STATUS_PEER_COMM</li> <li>• RF_STATUS_PEER_PRESENCE</li> <li>• RF_STATUS_REDUNDANCY_MODE_CHANGE</li> <li>• RF_STATUS_SWACT_INHIBIT</li> </ul> |

|                |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
|----------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| queue_priority | <p>(Optional) Priority level at which the script will be queued:</p> <ul style="list-style-type: none"> <li>• queue_priority low--Specifies that the script is to be queued at the lowest of the three priority levels.</li> <li>• queue_priority normal--Specifies that the script is to be queued at a priority level greater than low priority but less than high priority.</li> <li>• queue_priority high--Specifies that the script is to be queued at the highest of the three priority levels.</li> <li>• queue_priority last--Specifies that the script is to be queued at the lowest priority level.</li> </ul> <p>If more than one script is registered with the "queue_priority_last" argument set, these scripts will execute in the order in which the events are published.</p> <p><b>Note</b> The queue_priority argument specifies the queuing priority, but not the execution priority, of the script being registered.</p> <p>If this argument is not specified, the default queuing priority is normal.</p> |
| maxrun         | <p>(Optional) Maximum run time of the script (specified in SSSSSSSSS[.MMM] format, where SSSSSSSSS must be an integer representing seconds between 0 and 4294967295, inclusive, and where MMM must be an integer representing milliseconds between 0 and 999). If this argument is not specified, the default 20-second run-time limit is used.</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |
| nice           | <p>(Optional) Policy run-time priority setting. When the nice argument is set to 1, the policy is run at a run-time priority that is less than the default priority. The default value is 0.</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |

**Result String**

None

**Set\_cerrno**

No

**Event\_reqinfo**

```
"event_id %u event_type %u event_type_string {%s} %u event_pub_sec %u event_pub_msec %u"
"event {%s}"
```

| Event Type        | Description                                                                                                                                                 |
|-------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------|
| event_id          | Unique number that indicates the ID for this published event. Multiple policies may be run for the same event, and each policy will have the same event_id. |
| event_type        | Type of event.                                                                                                                                              |
| event_type_string | An ASCII string that represents the name of the event for this event type.                                                                                  |

| Event Type                                 | Description                                                                         |
|--------------------------------------------|-------------------------------------------------------------------------------------|
| <b>event_pub_sec</b> <b>event_pub_msec</b> | The time, in seconds and milliseconds, when the event was published to the EEM.     |
| <b>event</b>                               | RF progression or status event notification that caused this event to be published. |

## event\_register\_routing

Registers for an event that is triggered by the **event routing** command. These events are handled by the routing event detector to publish an event when route entries change in Routing Information Base (RIB) infrastructure. Use this Tcl command extension to run a routing policy for this script. The network IP address for the route to be monitored must be specified.

### Syntax

```
event_register_routing [tag ?] network ? length [ge|le|ne] [type add|remove|modify|all]
[protocol ?] [queue_priority normal|low|high|last] [maxrun ?] [nice {0 | 1}]
```

### Arguments

|          |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
|----------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| tag      | (Optional) String identifying a tag that can be used with the trigger Tcl command extension to support multiple event statements within a Tcl script.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |
| network  | Specifies the network IP address. The network number can be any valid IP address or prefix.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |
| length   | Specifies the length of the network mask in bits. The bit mask can be a number from 0 to 32. <ul style="list-style-type: none"> <li>• <b>ge</b> --(Optional) Specifies the minimum prefix length to be matched. The <b>ge</b> keyword represents greater than or equal to operator.</li> <li>• <b>le</b> --(Optional) Specifies the maximum prefix length to be matched. The <b>le</b> keyword represents the less than or equal to operator.</li> <li>• <b>ne</b> --(Optional) Specifies the prefix length not to be matched. The <b>ne</b> keyword represents not equal to operator.</li> </ul> <p>When <b>ge</b>, <b>le</b> and <b>ne</b> keywords are not configured, an exact match of network length is processed.</p> |
| type     | (Optional) Specifies the desired policy trigger. The type options are <b>add</b> , <b>remove</b> , <b>modify</b> , and <b>all</b> . The default is <b>all</b> .                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
| protocol | (Optional) Specifies the protocol value for the network being monitored. One of the following protocols can be used: <b>all</b> , <b>bgp</b> , <b>connected</b> , <b>eigrp</b> , <b>isis</b> , <b>iso-igrp</b> , <b>mobile</b> , <b>odr</b> , <b>ospf</b> , <b>rip</b> , and <b>static</b> . The default is <b>all</b> .                                                                                                                                                                                                                                                                                                                                                                                                     |

|                |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
|----------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| queue_priority | <p>(Optional) Priority level at which the script will be queued:</p> <ul style="list-style-type: none"> <li>• queue_priority low--Specifies that the script is to be queued at the lowest of the three priority levels.</li> <li>• queue_priority normal--Specifies that the script is to be queued at a priority level greater than low priority but less than high priority.</li> <li>• queue_priority high--Specifies that the script is to be queued at the highest of the three priority levels.</li> <li>• queue_priority last--Specifies that the script is to be queued at the lowest priority level.</li> </ul> <p>If more than one script is registered with the "queue_priority_last" argument set, these scripts will execute in the order in which the events are published.</p> <p><b>Note</b> The queue_priority argument specifies the queuing priority, but not the execution priority, of the script being registered.</p> <p>If this argument is not specified, the default queuing priority is normal.</p> |
| maxrun         | <p>(Optional) Maximum run time of the script (specified in SSSSSSSSS[.MMM] format, where SSSSSSSSS must be an integer representing seconds between 0 and 4294967295, inclusive, and where MMM must be an integer representing milliseconds between 0 and 999). If this argument is not specified, the default 20-second run-time limit is used.</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |
| nice           | <p>(Optional) Policy run-time priority setting. When the nice argument is set to 1, the policy is run at a run-time priority that is less than the default priority. The default value is 0.</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |

**Result String**

None

**Set\_cerrno**

No

**Event\_reqinfo**

"event\_id %u event\_type %u event\_type\_string {%s} %u event\_pub\_sec %u event\_pub\_msec %u"  
 "event\_severity {%s} %u network %u mask %u protocol %u lastgateway %u distance %u" "time\_sec %u  
 time\_msec %u metric %u lastinterface %u"

| Event Type        | Description                                                                                                                                                 |
|-------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------|
| event_id          | Unique number that indicates the ID for this published event. Multiple policies may be run for the same event, and each policy will have the same event_id. |
| event_type        | Type of event.                                                                                                                                              |
| event_type_string | An ASCII string that represents the name of the event for this event type.                                                                                  |

| Event Type                          | Description                                                                         |
|-------------------------------------|-------------------------------------------------------------------------------------|
| <b>event_pub_sec event_pub_msec</b> | The time, in seconds and milliseconds, when the event was published to the EEM.     |
| <b>event_severity</b>               | The severity of the event.                                                          |
| <b>network</b>                      | The network prefix in IP address format                                             |
| <b>mask</b>                         | The network mask in IP address format                                               |
| <b>protocol</b>                     | Type of network protocol.                                                           |
| <b>type</b>                         | Type of event to add, remove or modify.                                             |
| <b>lastgateway</b>                  | The last known gateway.                                                             |
| <b>distance</b>                     | The administrative distance.                                                        |
| <b>time_sec time_msec</b>           | Time of event in seconds and milliseconds, when the event was published to the EEM. |
| <b>metric</b>                       | Path metric.                                                                        |
| <b>lastinterface</b>                | The last known interface.                                                           |

## event\_register\_rpc

Registers for an event that is triggered by the EEM SSH Remote Procedure Call (RPC) command. These events are handled by the RPC event detector that screens for this event. Use this Tcl command extension to run a RPC policy for this script.

### Syntax

```
event_register_rpc [queue_priority {normal | low | high | last}] [maxrun <sec.msec>] [nice
{0 | 1}] [default <sec.msec>]
```

**Arguments**

|                |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
|----------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| queue_priority | <p>(Optional) Priority level at which the script will be queued:</p> <ul style="list-style-type: none"> <li>• queue_priority low--Specifies that the script is to be queued at the lowest of the three priority levels.</li> <li>• queue_priority normal--Specifies that the script is to be queued at a priority level greater than low priority but less than high priority.</li> <li>• queue_priority high--Specifies that the script is to be queued at the highest of the three priority levels.</li> <li>• queue_priority last--Specifies that the script is to be queued at the lowest priority level.</li> </ul> <p>If more than one script is registered with the "queue_priority_last" argument set, these scripts will execute in the order in which the events are published.</p> <p><b>Note</b> The queue_priority argument specifies the queuing priority, but not the execution priority, of the script being registered.</p> <p>If this argument is not specified, the default queuing priority is normal.</p> |
| maxrun         | <p>(Optional) Maximum run time of the script (specified in SSSSSSSSS[.MMM] format, where SSSSSSSSS must be an integer representing seconds between 0 and 4294967295, inclusive, and where MMM must be an integer representing milliseconds between 0 and 999). If this argument is not specified, the default 20-second run-time limit is used.</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |
| nice           | <p>(Optional) Policy run-time priority setting. When the nice argument is set to 1, the policy is run at a run-time priority that is less than the default priority. The default value is 0.</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |
| default        | <p>(Optional) The time period during which the CLI event detector waits for the policy to exit (specified in SSSSSSSSS[.MMM] format, where SSSSSSSSS must be an integer representing seconds between 0 and 4294967295, inclusive, and where MMM must be an integer representing milliseconds between 0 and 999). If the default time period expires before the policy exits, the default action will be executed. The default action is to run the command. If this argument is not specified, the default time period is set to 30 seconds.</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |

**Result String**

None

**Set\_cerrno**

No

**Event\_reqinfo**

```
"event_id %u event_type %u event_type_string {%s} event_pub_sec %u event_pub_msec %u arg %u"
```

| Event Type | Description |
|------------|-------------|
|------------|-------------|

|                                                                                                                                                                                                                                    |                                                                                                                                                             |
|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>event_id</b>                                                                                                                                                                                                                    | Unique number that indicates the ID for this published event. Multiple policies may be run for the same event, and each policy will have the same event_id. |
| <b>event_type</b>                                                                                                                                                                                                                  | Type of event.                                                                                                                                              |
| <b>event_type_string</b>                                                                                                                                                                                                           | An ASCII string that represents the name of the event for this event type.                                                                                  |
| <b>event_pub_sec event_pub_msec</b>                                                                                                                                                                                                | The time, in seconds and milliseconds, when the event was published to the EEM.                                                                             |
| <b>argc</b><br><b>arg0</b><br><b>arg1</b><br><b>arg2</b><br><b>arg3</b><br><b>arg4</b><br><b>arg6</b><br><b>arg7</b><br><b>arg8</b><br><b>arg9</b><br><b>arg10</b><br><b>arg11</b><br><b>arg12</b><br><b>arg13</b><br><b>arg14</b> | The parameters that are passed from the XML SOAP command to the script.                                                                                     |

## event\_register\_snmp

Registers for a Simple Network Management Protocol (SNMP) statistics event. Use this Tcl command extension to run a policy when a given counter specified by an SNMP object ID (oid) crosses a defined threshold.

### Syntax

```
event_register_snmp [tag ?] oid ? get_type exact|next
entry_op gt|ge|eq|ne|lt|le entry_val ?
entry_type value|increment|rate
[exit_comb or|and]
[exit_op gt|ge|eq|ne|lt|le] [exit_val ?]
[exit_type value|increment|rate]
[exit_time ?] poll_interval ? [average_factor ?]
[queue_priority low|normal|high|last]
[maxrun ?] [nice 0|1]
```

**Arguments**

|            |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |
|------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| tag        | (Optional) String identifying a tag that can be used with the trigger Tcl command extension to support multiple event statements within a Tcl script.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |
| oid        | (Mandatory) OID number of data element in SNMP dot notation (for example, 1.3.6.1.2.1.2.1.0). The types of OIDs allowed are: <ul style="list-style-type: none"> <li>• COUNTER_TYPE</li> <li>• COUNTER_64_TYPE</li> <li>• GAUGE_TYPE</li> <li>• INTEGER_TYPE</li> <li>• OCTET_PRIM_TYPE</li> <li>• OPAQUE_PRIM_TYPE</li> <li>• TIME_TICKS_TYPE</li> </ul>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
| entry_op   | (Mandatory) Entry comparison operator used to compare the current OID data value with the entry value; if true, an event will be raised and event monitoring will be disabled until exit criteria are met.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
| get_type   | (Mandatory) Type of SNMP get operation that needs to be applied to the OID specified. If the get_type argument is "exact," the value of the specified OID is retrieved; if the get_type argument is "next," the value of the lexicographical successor to the specified OID is retrieved.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |
| entry_val  | (Mandatory) Value with which the current oid data value should be compared to decide if the SNMP event should be raised.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
| entry-type | Specifies a type of operation to be applied to the object ID specified by the entry-val argument.<br>Value is defined as the actual value of the entry-val argument.<br><br>Increment uses the entry-val field as an incremental difference and the entry-val is compared with the difference between the current counter value and the value when the event was last triggered (or the first polled sample if this is a new event). A negative value checks the incremental difference for a counter that is decreasing.<br><br>Rate is defined as the average rate of change over a period of time. The time period is the average-factor value multiplied by the poll-interval value. At each poll interval the difference between the current sample and the previous sample is taken and recorded as an absolute value. An average of the previous average-factor value samples is taken to be the rate of change. |
| exit_comb  | (Optional) Exit combination operator used to indicate the combination of exit condition tests required to decide if the exit criteria are met so that the event monitoring can be reenabled. If it is "and," both exit value and exit time tests must be passed to meet the exit criteria. If it is "or," either exit value or exit time tests can be passed to meet the exit criteria. When exit_comb is "and," exit_op, and exit_val (exit_time) must exist. When exit_comb is "or," (exit_op and exit_val) or (exit_time) must exist.                                                                                                                                                                                                                                                                                                                                                                                |
| exit_op    | (Optional) Exit comparison operator used to compare the current oid data value with the exit value; if true, event monitoring for this event will be reenabled.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |



|                |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
|----------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| exit_val       | (Optional) Value with which the current oid data value should be compared to decide if the exit criteria are met.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
| exit-type      | <p>(Optional) Specifies a type of operation to be applied to the object ID specified by the exit-val argument. If not specified, the value is assumed.</p> <p>Value is defined as the actual value of the exit-val argument.</p> <p>Increment uses the exit-val field as an incremental difference and the exit-val is compared with the difference between the current counter value and the value when the event was last triggered (or the first polled sample if this is a new event). A negative value checks the incremental difference for a counter that is decreasing.</p> <p>Rate is defined as the average rate of change over a period of time. The time period is the average-factor value multiplied by the poll-interval value. At each poll interval the difference between the current sample and the previous sample is taken and recorded as an absolute value. An average of the previous average-factor value samples is taken to be the rate of change.</p>                                              |
| exit_time      | (Optional) Number of POSIX timer units after an event is raised when event monitoring will be enabled again. Specified in SSSSSSSSS[.MMM] format where SSSSSSSSS must be an integer number representing seconds between 0 and 4294967295, inclusive. MMM represents milliseconds and must be an integer number between 0 and 999.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
| poll_interval  | (Mandatory) Interval between consecutive polls in POSIX timer units. Currently the interval is forced to be at least 1 second (specified in SSSSSSSSS[.MMM] format, where SSSSSSSSS must be an integer representing seconds between 0 and 4294967295, inclusive, and where MMM must be an integer representing milliseconds between 0 and 999).                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
| average-factor | (Optional) Number in the range from 1 to 64 used to calculate the period used for rate-based calculations. The average-factor value is multiplied by the poll-interval value to derive the period in milliseconds. The minimum average factor value is 1.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |
| queue_priority | <p>(Optional) Priority level at which the script will be queued:</p> <ul style="list-style-type: none"> <li>• queue_priority low--Specifies that the script is to be queued at the lowest of the three priority levels.</li> <li>• queue_priority normal--Specifies that the script is to be queued at a priority level greater than low priority but less than high priority.</li> <li>• queue_priority high--Specifies that the script is to be queued at the highest of the three priority levels.</li> <li>• queue_priority last--Specifies that the script is to be queued at the lowest priority level.</li> </ul> <p>If more than one script is registered with the "queue_priority_last" argument set, these scripts will execute in the order in which the events are published.</p> <p><b>Note</b> The queue_priority argument specifies the queuing priority, but not the execution priority, of the script being registered.</p> <p>If this argument is not specified, the default queuing priority is normal.</p> |

|        |                                                                                                                                                                                                                                                                                                                                              |
|--------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| maxrun | (Optional) Maximum run time of the script (specified in SSSSSSSSS[.MMM] format, where SSSSSSSSS must be an integer representing seconds between 0 and 4294967295, inclusive, and where MMM must be an integer representing milliseconds between 0 and 999). If this argument is not specified, the default 20-second run-time limit is used. |
| nice   | (Optional) Policy run-time priority setting. When the nice argument is set to 1, the policy is run at a run-time priority that is less than the default priority. The default value is 0.                                                                                                                                                    |

**Result String**

None

**Set\_cerrno**

No

**Event\_reqinfo**

```
"event_id %u event_type %u event_type_string {%s} %u event_pub_sec %u event_pub_msec %u"
"event_severity {%s} oid {%s} val {%s} delta_val {%s}"
```

| Event Type                          | Description                                                                                                                                                 |
|-------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>event_id</b>                     | Unique number that indicates the ID for this published event. Multiple policies may be run for the same event, and each policy will have the same event_id. |
| <b>event_type</b>                   | Type of event.                                                                                                                                              |
| <b>event_type_string</b>            | An ASCII string that represents the name of the event for this event type.                                                                                  |
| <b>event_pub_sec event_pub_msec</b> | The time, in seconds and milliseconds, when the event was published to the EEM.                                                                             |
| <b>event_severity</b>               | SNMP event severity, which can be one of the following values: <ul style="list-style-type: none"> <li>• normal</li> <li>• minor</li> <li>• major</li> </ul> |
| <b>oid</b>                          | Object ID of data element, in SNMP dot notation.                                                                                                            |
| <b>val</b>                          | Value of the data element.                                                                                                                                  |
| <b>delta_val</b>                    | Delta value between the value of the policies.                                                                                                              |

## event\_register\_snmp\_notification

Registers for a Simple Network Management Protocol (SNMP) notification trap event. Use this Tcl command extension to run a policy when an SNMP trap with the specified SNMP object ID (oid) is encountered on a specific interface or address. The **snmp-server manager** CLI command must be enabled for the SNMP notifications to work using Tcl policies.

### Syntax

```
event_register_snmp_notification [tag ?] oid ? oid_val ?
op {gt|ge|eq|ne|lt|le}
[maxrun ?]
[src_ip_address ?]
[dest_ip_address ?]
[queue_priority {normal|low|high|last}]
[maxrun ?]
[nice {0|1}]
[default ?]
[direction {incoming|outgoing}]
[msg_op {drop|send}]
```

### Arguments

|         |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |
|---------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| tag     | (Optional) String identifying a tag that can be used with the trigger Tcl command extension to support multiple event statements within a Tcl script.                                                                                                                                                                                                                                                                                                                           |
| oid     | (Mandatory) OID number of the data element in SNMP dot notation (for example, 1.3.6.1.2.1.2.1.0). If the specified OID ends with a dot (.), then all OIDs that start with the OID number before the dot are matched. The types of OIDs allowed are: <ul style="list-style-type: none"> <li>• COUNTER_TYPE</li> <li>• COUNTER_64_TYPE</li> <li>• GAUGE_TYPE</li> <li>• INTEGER_TYPE</li> <li>• OCTET_PRIM_TYPE</li> <li>• OPAQUE_PRIM_TYPE</li> <li>• TIME_TICKS_TYPE</li> </ul> |
| oid_val | (Mandatory) OID value with which the current OID data value should be compared to decide if the SNMP event should be raised.                                                                                                                                                                                                                                                                                                                                                    |
| op      | (Mandatory) Comparison operator used to compare the current OID data value with the SNMP Protocol Data Unit (PDU) OID data value; if this is true, an event is raised.                                                                                                                                                                                                                                                                                                          |
| maxrun  | (Optional) Maximum run time of the script (specified in sssssss[.mmm] format, where sssssss must be an integer representing seconds between 0 and 31536000, inclusive, and where mmm must be an integer representing milliseconds between 0 and 999). If this argument is not specified, the default 20-second run-time limit is used.                                                                                                                                          |

|                 |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
|-----------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| src_ip_address  | (Optional) Source IP address where the SNMP notification trap originates. The default is all; it is set to receive SNMP notification traps from all IP addresses.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |
| dest_ip_address | (Optional) Destination IP address where the SNMP notification trap is sent. The default is all; it is set to receive SNMP traps from all destination IP addresses.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |
| queue_priority  | <p>(Optional) Priority level at which the script will be queued:</p> <ul style="list-style-type: none"> <li>• queue_priority low--Specifies that the script is to be queued at the lowest of the three priority levels.</li> <li>• queue_priority normal--Specifies that the script is to be queued at a priority level greater than low priority but less than high priority.</li> <li>• queue_priority high--Specifies that the script is to be queued at the highest of the three priority levels.</li> <li>• queue_priority last--Specifies that the script is to be queued at the lowest priority level.</li> </ul> <p>If more than one script is registered with the queue_priority_last argument set, these scripts will execute in the order in which the events are published.</p> <p><b>Note</b> The queue_priority argument specifies the queuing priority, but not the execution priority, of the script being registered.</p> <p>If this argument is not specified, the default queuing priority is normal.</p> |
| default         | (Optional) Specifies the time period in seconds during which the snmp notification event detector waits for the policy to exit. The time period is specified in ssssssss[.mmm] format, where ssssssss must be an integer representing seconds between 0 and 4294967295 and mmm must be an integer representing milliseconds between 0 and 999.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |
| nice            | (Optional) Policy run-time priority setting. When the nice argument is set to 1, the policy is run at a run-time priority that is less than the default priority. The default value is 0.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |
| direction       | (Optional) The direction of the incoming or outgoing SNMP trap or inform PDU to filter. The default value is incoming.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |
| msg_op          | (Optional) The action to be taken on the SNMP PDU (drop it or send it) once the event is triggered. The default value is send.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |

**Result String**

None

**Set\_cerrno**

No

**Event\_reqinfo**

```
"event_id %u event_type %u event_type_string {%s} %u event_pub_sec %u event_pub_msec %u
event_severity {%s}" "oid {%s} oid_val {%s} src_ip_addr {%s} dest_ip_addr {%s} x_x_x_x_x
(varbinds) {%s} trunc_vb_buf {%s} trap_oid {%s} enterprise_oid {%s} generic_trap %u
specific_trap %u"
```

| Event Type                          | Description                                                                                                                                                 |
|-------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>event_id</b>                     | Unique number that indicates the ID for this published event. Multiple policies may be run for the same event, and each policy will have the same event_id. |
| <b>event_type</b>                   | Type of event.                                                                                                                                              |
| <b>event_type_string</b>            | An ASCII string that represents the name of the event for this event type.                                                                                  |
| <b>event_pub_sec event_pub_msec</b> | The time, in seconds and milliseconds, when the event was published to the EEM.                                                                             |
| <b>oid</b>                          | An user specified object ID.                                                                                                                                |
| <b>oid_val</b>                      | An user specified object ID value.                                                                                                                          |
| <b>src_ip_addr</b>                  | The source IP address of the SNMP protocol data unit (PDU).                                                                                                 |
| <b>dest_ip_addr</b>                 | The destination IP address of the SNMP PDU.                                                                                                                 |
| <b>x_x_x_x_x (varbinds)</b>         | The SNMP PDU varbind information.                                                                                                                           |
| <b>trap_oid</b>                     | Indicates the trap OID value.                                                                                                                               |
| <b>enterprise_oid</b>               | Indicates the enterprise OID value.                                                                                                                         |
| <b>generic_trap</b>                 | Indicates one of a number of generic trap types. There are seven generic trap numbers zero to six.                                                          |
| <b>specific_trap</b>                | Indicates one of a number of specific trap codes.                                                                                                           |

## event\_register\_snmp\_object

Registers for a Simple Network Management Protocol (SNMP) object event. Use this Tcl command extension to replace the value when an SNMP with the specified SNMP-object ID (OID) is encountered on a specific interface or address.

### Syntax

```
event_register_snmp_object oid ?
type {int|uint|counter|counter64|gauge|ipv4||oid|string}
sync {yes|no}
skip {yes|no}
[istable {yes|no}]
[default ?]
[queue_priority {normal|low|high|last}]
[maxrun ?]
[nice {0|1}]
```

**Arguments**

|         |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |
|---------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| oid     | <p>(Mandatory) OID number of the data element in SNMP dot notation (for example, 1.3.6.1.2.1.2.1.0). If the specified OID ends with a dot (.), then all OIDs that start with the OID number before the dot are matched. The types of OIDs allowed are:</p> <ul style="list-style-type: none"> <li>• COUNTER_TYPE</li> <li>• COUNTER_64_TYPE</li> <li>• GAUGE_TYPE</li> <li>• INTEGER_TYPE</li> <li>• OCTET_PRIM_TYPE</li> <li>• OPAQUE_PRIM_TYPE</li> <li>• TIME_TICKS_TYPE</li> </ul>                                                                                                    |
| type    | (Mandatory) OID value type.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |
| sync    | <p>(Mandatory) A "yes" means that the EEM policy will be notified. If the applet set_exit_status or Tcl return value is 0, then SNMP will handle the request. If the return value is 1, SNMP will use the value provided by the policy for the get request and will not process the set request. A "no" means that EEM will not be notified and SNMP will handle the request.</p> <p>Only one OID can be associated with a synchronous policy. However, multiple synchronous policies can be registered for the same OID.</p>                                                             |
| skip    | Mandatory if the sync argument is "no" and should not exist if the sync argument is "yes." If the skip argument is "yes," it means that SNMP will handle the request. If the skip argument is "no," it means that SNMP will act as if the object does not exist.                                                                                                                                                                                                                                                                                                                          |
| istable | (Optional) A value of "no" means the OID is scalar object, and "yes" means the OID is table object.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |
| default | (Optional) The time period during which the SNMP Object event detector waits for the policy to exit (specified in ssssssss[.mmm] format, where ssssssss must be an integer representing seconds between 0 and 4294967295, inclusive, and where mmm must be an integer representing milliseconds between 0 and 999). If the default time period expires before the policy exits, the default action will be executed. The default action is to process the set or get request normally by SNMP subsystem. If this argument is not specified, the default time period is set to 30 seconds. |
| maxrun  | (Optional) Maximum run time of the script (specified in ssssssss[.mmm] format, where ssssssss must be an integer representing seconds between 0 and 31536000, inclusive, and where mmm must be an integer representing milliseconds between 0 and 999). If this argument is not specified, the default 20-second run-time limit is used.                                                                                                                                                                                                                                                  |

|                |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
|----------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| queue_priority | <p>(Optional) Priority level at which the script will be queued:</p> <ul style="list-style-type: none"> <li>• queue_priority low--Specifies that the script is to be queued at the lowest of the three priority levels.</li> <li>• queue_priority normal--Specifies that the script is to be queued at a priority level greater than low priority but less than high priority.</li> <li>• queue_priority high--Specifies that the script is to be queued at the highest of the three priority levels.</li> <li>• queue_priority last--Specifies that the script is to be queued at the lowest priority level.</li> </ul> <p>If more than one script is registered with the queue_priority_last argument set, these scripts will execute in the order in which the events are published.</p> <p><b>Note</b> The queue_priority argument specifies the queuing priority, but not the execution priority, of the script being registered.</p> <p>If this argument is not specified, the default queuing priority is normal.</p> |
| nice           | <p>(Optional) Policy run-time priority setting. When the nice argument is set to 1, the policy is run at a run-time priority that is less than the default priority. The default value is 0.</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |

**Result String**

None

**Set\_cerrno**

No

**Event\_reqinfo**

```
"event_id %u event_type %u event_type_string {%s} %u event_pub_sec %u event_pub_msec %u event_severity {%s}" "oid {%s} request {%s} request_type {%s} value %u"
```

| Event Type                   | Description                                                                                                                                                 |
|------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------|
| event_id                     | Unique number that indicates the ID for this published event. Multiple policies may be run for the same event, and each policy will have the same event_id. |
| event_type                   | Type of event.                                                                                                                                              |
| event_type_string            | An ASCII string that represents the name of the event for this event type.                                                                                  |
| event_pub_sec event_pub_msec | The time, in seconds and milliseconds, when the event was published to the EEM.                                                                             |
| event_severity               | The severity of the event.                                                                                                                                  |
| oid                          | The ID of the SNMP object in the received get or set request.                                                                                               |

| Event Type   | Description                                            |
|--------------|--------------------------------------------------------|
| request      | The get or set request type.                           |
| request_type | The type of request (exact or next).                   |
| value        | For set requests only. The value to set the object to. |

## event\_register\_syslog

Registers for a syslog event. Use this Tcl command extension to trigger a policy when a syslog message of a specific pattern is logged after a certain number of occurrences during a certain period of time.

### Syntax

```
event_register_syslog [tag ?] [occurs ?] [period ?] pattern ?
[priority all|emergencies|alerts|critical|errors|warnings|notifications|
informational|debugging|0|1|2|3|4|5|6|7]
[queue_priority low|normal|high|last]
[severity_fatal] [severity_critical] [severity_major]
[severity_minor] [severity_warning] [severity_notification]
[severity_normal] [severity_debugging]
[maxrun ?] [nice 0|1]
```

### Arguments

|          |                                                                                                                                                                                                                                                                                                                                                                                                                                            |
|----------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| tag      | (Optional) String identifying a tag that can be used with the trigger Tcl command extension to support multiple event statements within a Tcl script.                                                                                                                                                                                                                                                                                      |
| occurs   | (Optional) Number of occurrences before the event is raised; if not specified, the event is raised on the first occurrence. If specified, the value must be greater than 0.                                                                                                                                                                                                                                                                |
| period   | (Optional) Time interval, in seconds and milliseconds, during which the one or more occurrences must take place in order to raise an event (specified in SSSSSSSSS[.MMM] format where SSSSSSSSS must be an integer number representing seconds between 0 and 4294967295, inclusive, and where MMM represents milliseconds and must be an integer number between 0 and 999). If this argument is not specified, no period check is applied. |
| pattern  | (Mandatory) A regular expression used to perform syslog message pattern match. This argument is what the policy uses to identify the logged syslog message.                                                                                                                                                                                                                                                                                |
| priority | (Optional) The message priority to be screened. If this argument is specified, only messages that are at the specified logging priority level, or lower, are screened. If this argument is not specified, the default priority is 0.                                                                                                                                                                                                       |



|                |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
|----------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| queue_priority | <p>(Optional) Priority level at which the script will be queued:</p> <ul style="list-style-type: none"> <li>• queue_priority low--Specifies that the script is to be queued at the lowest of the three priority levels.</li> <li>• queue_priority normal--Specifies that the script is to be queued at a priority level greater than low priority but less than high priority.</li> <li>• queue_priority high--Specifies that the script is to be queued at the highest of the three priority levels.</li> <li>• queue_priority last--Specifies that the script is to be queued at the lowest priority level.</li> </ul> <p>If more than one script is registered with the "queue_priority_last" argument set, these scripts will execute in the order in which the events are published.</p> <p><b>Note</b> The queue_priority argument specifies the queuing priority, but not the execution priority, of the script being registered.</p> <p>If this argument is not specified, the default queuing priority is normal.</p> |
| maxrun         | <p>(Optional) Maximum run time of the script (specified in SSSSSSSSS[.MMM] format, where SSSSSSSSS must be an integer representing seconds between 0 and 4294967295, inclusive, and where MMM must be an integer representing milliseconds between 0 and 999). If this argument is not specified, the default 20-second run-time limit is used.</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |
| nice           | <p>(Optional) Policy run-time priority setting. When the nice argument is set to 1, the policy is run at a run-time priority that is less than the default priority. The default value is 0.</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |
| severity_xxx   | <p>(Optional) The event severity to be screened. If this argument is specified, only messages that are at the specified severity level are screened. See the table titled "Severity Level Mapping For Syslog Events" for the severity level mapping for syslog events.</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |

If multiple conditions are specified, the syslog event will be raised when all the conditions are matched.

**Table 219: Severity Level Mapping For Syslog Events**

| Severity Keyword      | Syslog Priority | Description                                          |
|-----------------------|-----------------|------------------------------------------------------|
| severity_fatal        | LOG_EMERG (0)   | System is unusable.                                  |
| severity_critical     | LOG_ALERT (1)   | Critical conditions, immediate attention required.   |
| severity_major        | LOG_CRIT (2)    | Major conditions.                                    |
| severity_minor        | LOG_ERR (3)     | Minor conditions.                                    |
| severity_warning      | LOG_WARNING (4) | Warning conditions.                                  |
| severity_notification | LOG_NOTICE (5)  | Basic notification, informational messages.          |
| severity_normal       | LOG_INFO (6)    | Normal event, indicates returning to a normal state. |
| severity_debugging    | LOG_DEBUG (7)   | Debugging messages.                                  |

**Result String**

None

**Set\_cerrno**

No

**Event\_reqinfo**

```
"event_id %u event_type %u event_type_string {%s} event_pub_sec %u event_pub_msec %u"
"msg {%s}"
```

| Event Type                          | Description                                                                                                                                                 |
|-------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>event_id</b>                     | Unique number that indicates the ID for this published event. Multiple policies may be run for the same event, and each policy will have the same event_id. |
| <b>event_type</b>                   | Type of event.                                                                                                                                              |
| <b>event_type_string</b>            | An ASCII string that represents the name of the event for this event type.                                                                                  |
| <b>event_pub_sec event_pub_msec</b> | The time, in seconds and milliseconds, when the event was published to the EEM.                                                                             |
| <b>msg</b>                          | The last syslog message that matches the pattern.                                                                                                           |

## event\_register\_timer

Creates a timer and registers for a timer event as both a publisher and a subscriber. Use this Tel command extension when there is a need to trigger a policy that is time specific or timer based. This event timer is both an event publisher and a subscriber. The publisher part indicates the conditions under which the named timer is to go off. The subscriber part identifies the name of the timer to which the event is subscribing.




---

**Note** Both the CRON and absolute time specifications work on local time.

---

**Syntax**

```
event_register_timer [tag ?] watchdog|countdown|absolute|cron
[name ?] [cron_entry ?]
[time ?]
[queue_priority low|normal|high|last] [maxrun ?]
[nice 0|1]
```

**Arguments**

|            |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |
|------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| tag        | (Optional) String identifying a tag that can be used with the trigger Tcl command extension to support multiple event statements within a Tcl script.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |
| watchdog   | (Mandatory) Watchdog timer.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |
| countdown  | (Mandatory) Countdown timer.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |
| absolute   | (Mandatory) Absolute timer.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |
| cron       | (Mandatory) CRON timer.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |
| name       | (Optional) Name of the timer.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |
| cron_entry | <p>(Optional) Must be specified if the CRON timer type is specified. Must not be specified if any other timer type is specified. A cron_entry is a partial UNIX crontab entry (the first five fields) as used with the UNIX CRON daemon.</p> <p>A cron_entry specification consists of a text string with five fields. The fields are separated by spaces. The fields represent the time and date when CRON timer events will be triggered. The fields are described in the table titled "Time and Date When CRON Events Will Be Triggered."</p> <p>Ranges of numbers are allowed. Ranges are two numbers separated with a hyphen. The specified range is inclusive. For example, 8-11 for an hour entry specifies execution at hours 8, 9, 10, and 11.</p> <p>A field may be an asterisk (*), which always stands for "first-last."</p> <p>Lists are allowed. A list is a set of numbers (or ranges) separated by commas. Examples: "1,2,5,9" and "0-4,8-12".</p> <p>Step values can be used in conjunction with ranges. Following a range with "/&lt;number&gt;" specifies skips of the number's value through the range. For example, "0-23/2" can be used in the hour field to specify an event that is triggered every other hour. Steps are also permitted after an asterisk, so if you want to say "every two hours", use "* /2".</p> <p>Names can also be used for the month and the day of week fields. Use the first three letters of the particular day or month (case does not matter). Ranges or lists of names are not allowed.</p> <p>The day on which a timer event is triggered can be specified by two fields: day of month and day of week. If both fields are restricted (that is, are not *), an event will be triggered when either field matches the current time. For example, "30 4 1,15 * 5" would cause an event to be triggered at 4:30 a.m. on the 1st and 15th of each month, plus every Friday.</p> <p>Instead of the first five fields, one of seven special strings may appear. These seven special strings are described in the table titled "Special Strings for cron_entry."</p> <p>Example 1: "0 0 1,15 * 1" would trigger an event at midnight on the 1st and 15th of each month, as well as on every Monday. To specify days by only one field, the other field should be set to *; "0 0 * * 1" would trigger an event at midnight only on Mondays.</p> <p>Example 2: "15 16 1 * *" would trigger an event at 4:15 p.m. on the first day of each month.</p> <p>Example 3: "0 12 * * 1-5" would trigger an event at noon on Monday through Friday of each week.</p> <p>Example 4: "@weekly" would trigger an event at midnight once a week on Sunday.</p> |

|                |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
|----------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| time           | (Optional) Must be specified if a timer type other than CRON is specified. Must not be specified if the CRON timer type is specified. For watchdog and countdown timers, the number of seconds and milliseconds until the timer expires; for the absolute timer, the calendar time of the expiration time. Time is specified in SSSSSSSSS[.MMM] format, where SSSSSSSSS must be an integer representing seconds between 0 and 4294967295, inclusive, and where MMM must be an integer representing milliseconds between 0 and 999. An absolute expiration date is the number of seconds and milliseconds since January 1, 1970. If the date specified has already passed, the timer expires immediately.                                                                                                                                                                                                                                                                                                                       |
| queue_priority | <p>(Optional) Priority level at which the script will be queued:</p> <ul style="list-style-type: none"> <li>• queue_priority low--Specifies that the script is to be queued at the lowest of the three priority levels.</li> <li>• queue_priority normal--Specifies that the script is to be queued at a priority level greater than low priority but less than high priority.</li> <li>• queue_priority high--Specifies that the script is to be queued at the highest of the three priority levels.</li> <li>• queue_priority last--Specifies that the script is to be queued at the lowest priority level.</li> </ul> <p>If more than one script is registered with the "queue_priority_last" argument set, these scripts will execute in the order in which the events are published.</p> <p><b>Note</b> The queue_priority argument specifies the queuing priority, but not the execution priority, of the script being registered.</p> <p>If this argument is not specified, the default queuing priority is normal.</p> |
| maxrun         | (Optional) Maximum run time of the script (specified in SSSSSSSSS[.MMM] format, where SSSSSSSSS must be an integer representing seconds between 0 and 4294967295, inclusive, and where MMM must be an integer representing milliseconds between 0 and 999). If this argument is not specified, the default 20-second run-time limit is used.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |
| nice           | (Optional) Policy run-time priority setting. When the nice argument is set to 1, the policy is run at a run-time priority that is less than the default priority. The default value is 0.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |

**Table 220: Time and Date When CRON Events Will Be Triggered**

| Field        | Allowed Values                                                                       |
|--------------|--------------------------------------------------------------------------------------|
| minute       | 0-59                                                                                 |
| hour         | 0-23                                                                                 |
| day of month | 1-31                                                                                 |
| month        | 1-12 (or names, see below)                                                           |
| day of week  | 0-7 (0 or 7 is Sun, or names; see the table titled "Special Strings for cron_entry") |

**Table 221: Special Strings for cron\_entry**

| String    | Meaning                            |
|-----------|------------------------------------|
| @yearly   | Trigger once a year, "0 0 1 1 *".  |
| @annually | Same as @yearly.                   |
| @monthly  | Trigger once a month, "0 0 1 * *". |
| @weekly   | Trigger once a week, "0 0 * * 0".  |
| @daily    | Trigger once a day, "0 0 * * *".   |
| @midnight | Same as @daily.                    |
| @hourly   | Trigger once an hour, "0 * * * *". |

**Result String**

None

**Set \_cerrno**

No

**Event\_reqinfo**

```
"event_id %u event_type %u event_type_string {%s} event_pub_sec %u event_pub_msec %u"
"timer_type %s timer_time_sec %ld timer_time_msec %ld"
"timer_remain_sec %ld timer_remain_msec %ld"
```

| Event Type                            | Description                                                                                                                                                 |
|---------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>event_id</b>                       | Unique number that indicates the ID for this published event. Multiple policies may be run for the same event, and each policy will have the same event_id. |
| <b>event_type</b>                     | Type of event.                                                                                                                                              |
| <b>event_type_string</b>              | An ASCII string that represents the name of the event for this event type.                                                                                  |
| <b>event_pub_sec event_pub_msec</b>   | The time, in seconds and milliseconds, when the event was published to the EEM.                                                                             |
| <b>timer_type</b>                     | Type of the timer. Can be one of the following: <ul style="list-style-type: none"> <li>• watchdog</li> <li>• countdown</li> <li>• absolute</li> </ul>       |
| <b>timer_time_sec timer_time_msec</b> | Time when the timer expired.                                                                                                                                |

| Event Type                            | Description                                    |
|---------------------------------------|------------------------------------------------|
| timer_remain_sec<br>timer_remain_msec | The remaining time before the next expiration. |

**See Also**

event\_register\_timer\_subscriber

## event\_register\_timer\_subscriber

Registers for a timer event as a subscriber. Use this Tcl command extension to identify the name of the timer to which the event timer, as a subscriber, wants to subscribe. The event timer depends on another policy or another process to actually manipulate the timer. For example, let policyB act as a timer subscriber policy, but policyA (although it does not need to be a timer policy) uses register\_timer, timer\_arm, or timer\_cancel Tcl command extensions to manipulate the timer referenced in policyB.

**Syntax**

```
event_register_timer_subscriber watchdog|countdown|absolute|cron
name ? [queue_priority low|normal|high|last] [maxrun ?] [nice 0|1]
```

**Arguments**

|           |                                |
|-----------|--------------------------------|
| watchdog  | (Mandatory) Watchdog timer.    |
| countdown | (Mandatory) Countdown timer.   |
| absolute  | (Mandatory) Absolute timer.    |
| cron      | (Mandatory) CRON timer.        |
| name      | (Mandatory) Name of the timer. |

|                |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
|----------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| queue_priority | <p>(Optional) Priority level at which the script will be queued:</p> <ul style="list-style-type: none"> <li>• queue_priority low--Specifies that the script is to be queued at the lowest of the three priority levels.</li> <li>• queue_priority normal--Specifies that the script is to be queued at a priority level greater than low priority but less than high priority.</li> <li>• queue_priority high--Specifies that the script is to be queued at the highest of the three priority levels.</li> <li>• queue_priority last--Specifies that the script is to be queued at the lowest priority level.</li> </ul> <p>If more than one script is registered with the "queue_priority_last" argument set, these scripts will execute in the order in which the events are published.</p> <p><b>Note</b> The queue_priority argument specifies the queuing priority, but not the execution priority, of the script being registered.</p> <p>If this argument is not specified, the default queuing priority is normal.</p> |
| maxrun         | <p>(Optional) Maximum run time of the script (specified in SSSSSSSSS[.MMM] format, where SSSSSSSSS must be an integer representing seconds between 0 and 4294967295, inclusive, and where MMM must be an integer representing milliseconds between 0 and 999). If this argument is not specified, the default 20-second run-time limit is used.</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |
| nice           | <p>(Optional) Policy run-time priority setting. When the nice argument is set to 1, the policy is run at a run-time priority that is less than the default priority. The default value is 0.</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |



**Note** An EEM policy that registers for a timer event or a counter event can act as both publisher and subscriber.

**Result String**

None

**Set\_cerrno**

No

**Event\_reqinfo**

```
"event_id %u event_type %u event_type_string {%s} event_pub_sec %u event_pub_msec %u"
"timer_type %s timer_time_sec %ld timer_time_msec %ld"
"timer_remain_sec %ld timer_remain_msec %ld"
```

| Event Type | Description                                                                                                                                                 |
|------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------|
| event_id   | Unique number that indicates the ID for this published event. Multiple policies may be run for the same event, and each policy will have the same event_id. |
| event_type | Type of event.                                                                                                                                              |

| Event Type                                    | Description                                                                                                                                           |
|-----------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>event_type_string</b>                      | An ASCII string that represents the name of the event for this event type.                                                                            |
| <b>event_pub_sec event_pub_msec</b>           | The time, in seconds and milliseconds, when the event was published to the EEM.                                                                       |
| <b>timer_type</b>                             | Type of the timer. Can be one of the following: <ul style="list-style-type: none"> <li>• watchdog</li> <li>• countdown</li> <li>• absolute</li> </ul> |
| <b>timer_time_sec timer_time_msec</b>         | Time when the timer expired.                                                                                                                          |
| <b>timer_remain_sec<br/>timer_remain_msec</b> | The remaining time before the next expiration.                                                                                                        |

**See Also**

event\_register\_timer

## event\_register\_track

Registers for a report event from the Cisco IOS Object Tracking subsystem. Use this Tcl command extension to trigger a policy on the basis of a Cisco IOS Object Tracking subsystem report for a specified object number.

**Syntax**

```
event_register_track ? [tag ?] [state up|down|any] [queue_priority low|normal|high|last]
[maxrun ?]
[nice 0|1]
```

**Arguments**

|                         |                                                                                                                                                                                                                                                                                                                                                                                                                                                             |
|-------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| ? (represents a number) | (Mandatory) Tracked object number in the range from 1 to 500, inclusive.                                                                                                                                                                                                                                                                                                                                                                                    |
| tag                     | (Optional) String identifying a tag that can be used with the trigger Tcl command extension to support multiple event statements within a Tcl script.                                                                                                                                                                                                                                                                                                       |
| state                   | (Optional) Specifies that the tracked object transition will cause an event to be raised. If <b>up</b> is specified, an event will be raised when the tracked object transitions from a down state to an up state. If <b>down</b> is specified, an event will be raised when the tracked object transitions from an up state to a down state. If <b>any</b> is specified, an event will be raised when the tracked object transitions to or from any state. |



|                |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
|----------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| queue_priority | <p>(Optional) Priority level at which the script will be queued:</p> <ul style="list-style-type: none"> <li>• queue_priority low--Specifies that the script is to be queued at the lowest of the three priority levels.</li> <li>• queue_priority normal--Specifies that the script is to be queued at a priority level greater than low priority but less than high priority.</li> <li>• queue_priority high--Specifies that the script is to be queued at the highest of the three priority levels.</li> <li>• queue_priority last--Specifies that the script is to be queued at the lowest priority level.</li> </ul> <p>If more than one script is registered with the "queue_priority_last" argument set, these scripts will execute in the order in which the events are published.</p> <p><b>Note</b> The queue_priority argument specifies the queuing priority, but not the execution priority, of the script being registered.</p> <p>If this argument is not specified, the default queuing priority is normal.</p> |
| maxrun         | <p>(Optional) Maximum run time of the script (specified in SSSSSSSSS[.MMM] format, where SSSSSSSSS must be an integer representing seconds between 0 and 4294967295, inclusive, and where MMM must be an integer representing milliseconds between 0 and 999). If this argument is not specified, the default 20-second run-time limit is used.</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |
| nice           | <p>(Optional) Policy run-time priority setting. When the nice argument is set to 1, the policy is run at a run-time priority that is less than the default priority. The default value is 0.</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |

If an optional argument is not specified, the event matches all possible values of the argument.

**Result String**

None

**Set\_cerrno**

No

**Event\_reqinfo**

```
"event_id %u event_type %u event_type_string {%s} %u event_pub_sec %u event_pub_msec %u"
"track_number {%u} track_state {%s}"
```

| Event Type        | Description                                                                                                                                                 |
|-------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------|
| event_id          | Unique number that indicates the ID for this published event. Multiple policies may be run for the same event, and each policy will have the same event ID. |
| event_type        | Type of event.                                                                                                                                              |
| event_type_string | An ASCII string that represents the name of the event for this event type.                                                                                  |

| Event Type                   | Description                                                                            |
|------------------------------|----------------------------------------------------------------------------------------|
| event_pub_sec event_pub_msec | The time, in seconds and milliseconds, when the event was published to the EEM.        |
| track_number                 | Number of the tracked object that caused the event to be triggered.                    |
| track_state                  | State of the tracked object when the event was triggered; valid states are up or down. |

## event\_register\_wdsysmon

Registers for a Watchdog system monitor event. Use this Tcl command extension to register for a composite event which is a combination of several subevents or conditions. For example, you can use this command to register for the combination of conditions wherein the CPU usage of a certain process is over 80 percent and the memory used by the process is greater than 50 percent of its initial allocation. This Tcl command extension is supported only in Software Modularity images.

### Syntax

```
event_register_wdsysmon [tag ?] [timewin ?]
[sub12_op and|or|andnot]
[sub23_op and|or|andnot]
[sub34_op and|or|andnot]
[sub1 subevent-description]
[sub2 subevent-description]
[sub3 subevent-description]
[sub4 subevent-description] [node ?]
[queue_priority low|normal|high|last]
[maxrun ?] [nice 0|1]
```

Each argument is position independent.



**Note** Operator definitions: and (logical and operation), or (logical or operation), andnot (logical and not operation). For example, "sub12\_op and" is defined as raise an event when subevent 1 and subevent 2 are true; "sub23\_op or" is defined as raise an event when the condition specified in sub12\_op is true or subevent 3 is true. The logic can be diagrammed using: if (((sub1 sub12\_op sub2) sub23\_op sub3) sub34\_op sub4) is TRUE, raise event

### Arguments

|         |                                                                                                                                                                                                                                                                                                                                 |
|---------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| tag     | (Optional) String identifying a tag that can be used with the trigger Tcl command extension to support multiple event statements within a Tcl script.                                                                                                                                                                           |
| timewin | (Optional) Time window within which all of the subevents have to occur in order for an event to be generated (specified in SSSSSSSSS[.MMM] format, where SSSSSSSSSS must be an integer representing seconds between 0 and 4294967295, inclusive, and where MMM must be an integer representing milliseconds between 0 and 999). |

|                      |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
|----------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| sub12_op             | (Optional) Combination operator for comparison between subevent 1 and subevent 2.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
| sub23_op             | (Optional) Combination operator for comparison between subevent 1 and 2 and subevent 3.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |
| sub34_op             | (Optional) Combination operator for comparison between subevent 1 and 2 and subevent 3 and subevent 4.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |
| sub1                 | (Optional) Indicates that subevent 1 is specified.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |
| subevent-description | (Optional) Syntax for the subevent.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |
| sub2                 | (Optional) Indicates that subevent 2 is specified.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |
| sub3                 | (Optional) Indicates that subevent 3 is specified.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |
| sub4                 | (Optional) Indicates that subevent 4 is specified.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |
| node                 | <p>(Optional) The node name to be monitored for deadlock conditions is a string that consists of the word "node" followed by two fields separated by a slash character using the following format:</p> <pre>node&lt;slot-number&gt;/&lt;cpu-number&gt;</pre> <p>The slot-number is the hardware slot number. The cpu-number is the hardware CPU number. For example, the SP CPU in a Supervisor card on a Cisco Catalyst 6500 series switch located in slot 0 would be specified as node0/0. The RP CPU in a Supervisor card on a Cisco Catalyst 6500 series switch located in slot 0 would be addressed as node0/1. If the node argument is not specified, the default node specification is the local node on which the registration is done.</p>                                                                                                                                                                                                                                                                            |
| queue_priority       | <p>(Optional) Priority level at which the script will be queued:</p> <ul style="list-style-type: none"> <li>• queue_priority low--Specifies that the script is to be queued at the lowest of the three priority levels.</li> <li>• queue_priority normal--Specifies that the script is to be queued at a priority level greater than low priority but less than high priority.</li> <li>• queue_priority high--Specifies that the script is to be queued at the highest of the three priority levels.</li> <li>• queue_priority last--Specifies that the script is to be queued at the lowest priority level.</li> </ul> <p>If more than one script is registered with the "queue_priority_last" argument set, these scripts will execute in the order in which the events are published.</p> <p><b>Note</b> The queue_priority argument specifies the queuing priority, but not the execution priority, of the script being registered.</p> <p>If this argument is not specified, the default queuing priority is normal.</p> |

|        |                                                                                                                                                                                                                                                                                                                                              |
|--------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| maxrun | (Optional) Maximum run time of the script (specified in SSSSSSSSS[.MMM] format, where SSSSSSSSS must be an integer representing seconds between 0 and 4294967295, inclusive, and where MMM must be an integer representing milliseconds between 0 and 999). If this argument is not specified, the default 20-second run-time limit is used. |
| nice   | (Optional) Policy run-time priority setting. When the nice argument is set to 1, the policy is run at a run-time priority that is less than the default priority. The default value is 0.                                                                                                                                                    |

### Subevents

The syntax of subevent descriptions can be one of seven cases.

For arguments in subevent description, the following constraints apply on the value of number arguments:

- For `dispatch_mgr`, `val` must be an integer between 0 and 4294967295, inclusive.
- For `cpu_proc` and `cpu_tot`, `val` must be an integer between 0 and 100, inclusive.
- For `mem_proc`, `mem_tot_avail`, and `mem_tot_used`, if `is_percent` is FALSE, `val` must be an integer between 0 and 4294967295, inclusive.

1. `deadlock procname ?`

### Arguments

|          |                                                                                                                                                                                   |
|----------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| procname | (Mandatory) A regular expression that specifies the process name that you wish to monitor for deadlock conditions. This subevent will ignore the time window even if it is given. |
|----------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|

2. `dispatch_mgr [procname ?] [op gt|ge|eq|ne|lt|le] [val ?] [period ?]`

### Arguments

|          |                                                                                                                                                                                                                                                                                                                                                             |
|----------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| procname | (Optional) A regular expression that specifies the process name that you wish to monitor for <code>dispatch_manager</code> status.                                                                                                                                                                                                                          |
| op       | (Optional) Comparison operator used to compare the collected number of events with the specified value; if true, an event will be raised.                                                                                                                                                                                                                   |
| val      | (Optional) The value with which the number of events that have occurred should be compared.                                                                                                                                                                                                                                                                 |
| period   | (Optional) The time period for the number of events that have occurred (specified in SSSSSSSSS[.MMM] format, where SSSSSSSSS must be an integer representing seconds between 0 and 4294967295, inclusive, and where MMM must be an integer representing milliseconds between 0 and 999). If this argument is not specified, the most recent sample is used. |

3. `cpu_proc [procname ?] [op gt|ge|eq|ne|lt|le] [val ?] [period ?]`

**Arguments**

|          |                                                                                                                                                                                                                                                                                                                                                         |
|----------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| procname | (Optional) A regular expression that specifies the process name that you wish to monitor for CPU utilization conditions.                                                                                                                                                                                                                                |
| op       | (Optional) Comparison operator used to compare the collected CPU usage sample percentage with the specified percentage value; if true, an event will be raised.                                                                                                                                                                                         |
| val      | (Optional) The percentage value with which the average CPU usage during the sample period should be compared.                                                                                                                                                                                                                                           |
| period   | (Optional) The time period for averaging the collection of samples (specified in SSSSSSSSS[.MMM] format, where SSSSSSSSS must be an integer representing seconds between 0 and 4294967295, inclusive, and where MMM must be an integer representing milliseconds between 0 and 999). If this argument is not specified, the most recent sample is used. |

4. cpu\_tot [op gt|ge|eq|ne|lt|le] [val ?] [period ?]

**Arguments**

|        |                                                                                                                                                                                                                                                                                                                                                         |
|--------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| op     | (Optional) Comparison operator used to compare the collected total system CPU usage sample percentage with the specified percentage value; if true, an event will be raised.                                                                                                                                                                            |
| val    | (Optional) The percentage value with which the average CPU usage during the sample period should be compared.                                                                                                                                                                                                                                           |
| period | (Optional) The time period for averaging the collection of samples (specified in SSSSSSSSS[.MMM] format, where SSSSSSSSS must be an integer representing seconds between 0 and 4294967295, inclusive, and where MMM must be an integer representing milliseconds between 0 and 999). If this argument is not specified, the most recent sample is used. |

5. mem\_proc [procname ?] [op gt|ge|eq|ne|lt|le] [val ?] [is\_percent TRUE|FALSE] [period ?]

**Arguments**

|            |                                                                                                                                                                                                                                                                                                                                                                           |
|------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| procname   | (Optional) A regular expression that specifies the process name that you wish to monitor for memory usage.                                                                                                                                                                                                                                                                |
| op         | (Optional) Comparison operator used to compare the collected memory used with the specified value; if true, an event will be raised.                                                                                                                                                                                                                                      |
| val        | (Optional) A percentage or an absolute value specified in kilobytes. A percentage represents the difference between the oldest sample in the specified time period and the latest sample. If memory usage has increased from 150 KB to 300 KB within the time period, the percentage increase is 100. This is the value with which the measured value should be compared. |
| is_percent | (Optional) If TRUE, the percentage value is collected and compared. Otherwise, the absolute value is collected and compared.                                                                                                                                                                                                                                              |

|        |                                                                                                                                                                                                                                                                                                                                                                                                                                                       |
|--------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| period | (Optional) If is_percent is set to TRUE, the time period for the percentage to be computed. Otherwise, the time period for the collection samples to be averaged (specified in SSSSSSSSS[.MMM] format, where SSSSSSSSS must be an integer representing seconds between 0 and 4294967295, inclusive, and where MMM must be an integer representing milliseconds between 0 and 999). If this argument is not specified, the most recent sample is used. |
|--------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|

```
6. mem_tot_avail [op gt|ge|eq|ne|lt|le] [val ?] [is_percent TRUE|FALSE] [period ?]
```

### Arguments

|            |                                                                                                                                                                                                                                                                                                                                                                                                                                                       |
|------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| op         | (Optional) Comparison operator used to compare the collected available memory with the specified value; if true, an event will be raised.                                                                                                                                                                                                                                                                                                             |
| val        | (Optional) A percentage or an absolute value specified in kilobytes. A percentage represents the difference between the oldest sample in the specified time period and the latest sample. If available memory usage has decreased from 300 KB to 150 KB within the time period, the percentage decrease is 50. This is the value with which the measured value should be compared.                                                                    |
| is_percent | (Optional) If TRUE, the percentage value is collected and compared. Otherwise, the absolute value is collected and compared.                                                                                                                                                                                                                                                                                                                          |
| period     | (Optional) If is_percent is set to TRUE, the time period for the percentage to be computed. Otherwise, the time period for the collection samples to be averaged (specified in SSSSSSSSS[.MMM] format, where SSSSSSSSS must be an integer representing seconds between 0 and 4294967295, inclusive, and where MMM must be an integer representing milliseconds between 0 and 999). If this argument is not specified, the most recent sample is used. |

```
7. mem_tot_used [op gt|ge|eq|ne|lt|le] [val ?] [is_percent TRUE|FALSE] [period ?]
```

### Arguments

|            |                                                                                                                                                                                                                                                                                                                                                                           |
|------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| op         | (Optional) Comparison operator used to compare the collected used memory with the specified value; if true, an event will be raised.                                                                                                                                                                                                                                      |
| val        | (Optional) A percentage or an absolute value specified in kilobytes. A percentage represents the difference between the oldest sample in the specified time period and the latest sample. If memory usage has increased from 150 KB to 300 KB within the time period, the percentage increase is 100. This is the value with which the measured value should be compared. |
| is_percent | (Optional) If TRUE, the percentage value is collected and compared. Otherwise, the absolute value is collected and compared.                                                                                                                                                                                                                                              |

|        |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |
|--------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| period | <p>(Optional) If is_percent is set to TRUE, the time period for the percentage to be computed. Otherwise, the time period for the collection samples to be averaged (specified in SSSSSSSSS[.MMM] format, where SSSSSSSSS must be an integer representing seconds between 0 and 4294967295, inclusive, and where MMM must be an integer representing milliseconds between 0 and 999). If this argument is not specified, the most recent sample is used.</p> <p><b>Note</b> This argument is mandatory if is_percent is set to TRUE; otherwise, it is optional.</p> |
|--------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|

**Result String**

None

**Set\_cerrno**

No

**Event\_reqinfo**

```
"event_id %u event_type %u event_type_string {%s} %u event_pub_sec %u event_pub_msec %u"
"num_subs %u"
```

| Event Type                   | Description                                                                                                                                                 |
|------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------|
| event_id                     | Unique number that indicates the ID for this published event. Multiple policies may be run for the same event, and each policy will have the same event_id. |
| event_type                   | Type of event.                                                                                                                                              |
| event_type_string            | An ASCII string that represents the name of the event for this event type.                                                                                  |
| event_pub_sec event_pub_msec | The time, in seconds and milliseconds, when the event was published to the EEM.                                                                             |
| num_subs                     | Subevent number.                                                                                                                                            |

Where the subevent info string is for a deadlock subevent:

```
"{type %s num_entries %u entries {entry 1, entry 2, ...}}"
```

| Subevent Type | Description                                           |
|---------------|-------------------------------------------------------|
| type          | Type of wdsysmon subevent.                            |
| num_entries   | Number of processes and threads in the deadlock.      |
| entries       | Information of processes and threads in the deadlock. |

Where each entry is:

```
"{node {%s} procname {%s} pid %u tid %u state %s b_node %s b_procname %s b_pid %u
b_tid %u}"
```

Assume that the entry describes the scenario in which Process A thread m is blocked on process B thread n:

| Subevent Type     | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |
|-------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>node</b>       | Name of the node that process A thread m is on.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |
| <b>procname</b>   | Name of process A.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |
| <b>pid</b>        | Process ID of process A.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |
| <b>tid</b>        | Thread ID of process A thread m.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |
| <b>state</b>      | Thread state of process A thread m. Can be one of the following: <ul style="list-style-type: none"> <li>• STATE_CONDVAR</li> <li>• STATE_DEAD</li> <li>• STATE_INTR</li> <li>• STATE_JOIN</li> <li>• STATE_MUTEX</li> <li>• STATE_NANOSLEEP</li> <li>• STATE_READY</li> <li>• STATE_RECEIVE</li> <li>• STATE_REPLY</li> <li>• STATE_RUNNING</li> <li>• STATE_SEM</li> <li>• STATE_SEND</li> <li>• STATE_SIGSUSPEND</li> <li>• STATE_SIGWAITINFO</li> <li>• STATE_STACK</li> <li>• STATE_STOPPED</li> <li>• STATE_WAITPAGE</li> <li>• STATE_WAITTHREAD</li> </ul> |
| <b>b_node</b>     | Name of the node that process B thread is on.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |
| <b>b_procname</b> | Name of process B.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |
| <b>b_pid</b>      | Process ID of process B.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |



| Subevent Type | Description                                                                                              |
|---------------|----------------------------------------------------------------------------------------------------------|
| <b>b_tid</b>  | Thread ID of process B thread n; 0 means that process A thread m is blocked on all threads of process B. |

**For dispatch\_mgr Subevent**

```
"{type %s node {%s} procname {%s} pid %u value %u sec %ld msec %ld}"
```

| Subevent Type   | Description                                                                                                                                                                                                                                                                                                                                                                                                           |
|-----------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>type</b>     | Type of wdsysmon subevent.                                                                                                                                                                                                                                                                                                                                                                                            |
| <b>node</b>     | Name of the node that the POSIX process is on.                                                                                                                                                                                                                                                                                                                                                                        |
| <b>procname</b> | POSIX process name for this subevent.                                                                                                                                                                                                                                                                                                                                                                                 |
| <b>pid</b>      | POSIX process ID for this subevent.<br><b>Note</b> The three fields above describe the owner process of this dispatch manager.                                                                                                                                                                                                                                                                                        |
| <b>value</b>    | If the <b>sec</b> and <b>msec</b> variables are specified as 0 or are unspecified in the event registration Tcl command extension, the number of events processed by the dispatch manager is in the latest sample. If a time window is specified and is greater than zero in the event registration Tcl command extension, the total number of events processed by this dispatch manager is in the given time window. |
| <b>sec msec</b> | If the <b>sec</b> and <b>msec</b> variables are specified as 0 or are unspecified in the event registration Tcl command extension, they are both 0. If a time window is specified and is greater than zero in the event registration Tcl command extension, the <b>sec</b> and <b>msec</b> variables are the actual time difference between the time stamps of the oldest and latest samples in this time window.     |

**For cpu\_proc Subevent**

```
"{type %s node {%s} procname {%s} pid %u value %u sec %ld msec %ld}"
```

| Subevent Type   | Description                                                                                                                              |
|-----------------|------------------------------------------------------------------------------------------------------------------------------------------|
| <b>type</b>     | Type of wdsysmon subevent.                                                                                                               |
| <b>node</b>     | Name of the node that the POSIX process is on.                                                                                           |
| <b>procname</b> | POSIX process name for this subevent.                                                                                                    |
| <b>pid</b>      | POSIX process ID for this subevent.<br><b>Note</b> The three fields above describe the process whose CPU utilization is being monitored. |

| Subevent Type   | Description                                                                                                                                                                                                                                                                                                                                                                                                       |
|-----------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>value</b>    | If the <b>sec</b> and <b>msec</b> variables are specified as 0 or are unspecified in the event registration Tcl command extension, the process CPU utilization is in the latest sample. If a time window is specified and is greater than zero in the event registration Tcl command extension, the averaged process CPU utilization is in the given time window.                                                 |
| <b>sec msec</b> | If the <b>sec</b> and <b>msec</b> variables are specified as 0 or are unspecified in the event registration Tcl command extension, they are both 0. If a time window is specified and is greater than zero in the event registration Tcl command extension, the <b>sec</b> and <b>msec</b> variables are the actual time difference between the time stamps of the oldest and latest samples in this time window. |

#### For cpu\_tot Subevent

```
"{type %s node {%s} value %u sec %ld msec %ld}"
```

| Subevent Type   | Description                                                                                                                                                                                                                                                                                                                                                                                                       |
|-----------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>type</b>     | Type of wdsysmon subevent.                                                                                                                                                                                                                                                                                                                                                                                        |
| <b>node</b>     | Name of the node on which the total CPU utilization is being monitored.                                                                                                                                                                                                                                                                                                                                           |
| <b>value</b>    | If the <b>sec</b> and <b>msec</b> variables are specified as 0 or are unspecified in the event registration Tcl command extension, the total CPU utilization is in the latest sample. If a time window is specified and is greater than zero in the event registration Tcl command extension, the averaged total CPU utilization is in the given time window.                                                     |
| <b>sec msec</b> | If the <b>sec</b> and <b>msec</b> variables are specified as 0 or are unspecified in the event registration Tcl command extension, they are both 0. If a time window is specified and is greater than zero in the event registration Tcl command extension, the <b>sec</b> and <b>msec</b> variables are the actual time difference between the time stamps of the oldest and latest samples in this time window. |

#### For mem\_proc Subevent

```
"{type %s node {%s} procname {%s} pid %u is_percent %s value %u diff %d sec %ld msec %ld}"
```

| Subevent Type   | Description                                                                                    |
|-----------------|------------------------------------------------------------------------------------------------|
| <b>type</b>     | Type of wdsysmon subevent.                                                                     |
| <b>node</b>     | Name of the node that the POSIX process is on.                                                 |
| <b>procname</b> | POSIX process name for this subevent.                                                          |
| <b>pid</b>      | POSIX process ID for this subevent.                                                            |
|                 | <b>Note</b> The three fields above describe the process whose memory usage is being monitored. |

| Subevent Type     | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |
|-------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>is_percent</b> | Can be either TRUE or FALSE. TRUE means that the value is a percentage value; FALSE means that the value is an absolute value (may be an averaged value).                                                                                                                                                                                                                                                                                                                                                                               |
| <b>value</b>      | If the <b>sec</b> and <b>msec</b> variables are specified as 0 or are unspecified in the event registration Tcl command extension, the process used memory is in the latest sample. If a time window is specified and is greater than zero in the event registration Tcl command extension, the averaged process used memory utilization is in the given time window.                                                                                                                                                                   |
| Subevent Type     | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |
| <b>diff</b>       | If the <b>sec</b> and <b>msec</b> variables are specified as 0 or are unspecified in the event registration Tcl command extension, the <b>diff</b> is the percentage difference between the first process used memory sample ever collected and the latest process used memory sample. If a time window is specified and is greater than zero in the event registration Tcl command extension, the <b>diff</b> is the percentage difference between the oldest and latest process used memory utilization in the specified time window. |
| <b>sec msec</b>   | If the <b>sec</b> and <b>msec</b> variables are specified as 0 or are unspecified in the event registration Tcl command extension, they are both 0. If a time window is specified and is greater than zero in the event registration Tcl command extension, the <b>sec</b> and <b>msec</b> variables are the actual time difference between the time stamps of the oldest and latest samples in this time window.                                                                                                                       |

If the **is\_percent** argument is FALSE, and the **sec** and **msec** arguments are specified as 0 or are unspecified in the event registration Tcl command extension:

- **value** is the process used memory in the latest sample.
- **diff** is 0.
- **sec** and **msec** are both 0.

If the **is\_percent** argument is FALSE, and a time window is specified as greater than zero in the event registration Tcl command extension:

- **value** is the averaged process used memory sample value in the specified time window.
- **diff** is 0.
- **sec** and **msec** are both the actual time difference between the time stamps of the oldest and latest samples in this time window.

If the **is\_percent** argument is TRUE, and a time window is specified as greater than zero in the event registration Tcl command extension:

- **value** is 0.
- **diff** is the percentage difference between the oldest and latest process used memory samples in the specified time window.
- **sec** and **msec** are the actual time difference between the time stamps of the oldest and latest process used memory samples in this time window.

If the **is\_percent** argument is TRUE, and the **sec** and **msec** arguments are specified as 0 or are unspecified in the event registration Tcl command extension:

- **value** is 0.
- **diff** is the percentage difference between the first process used memory sample ever collected and the latest process used memory sample.
- **sec** and **msec** are the actual time difference between the time stamps of the first process used memory sample ever collected and the latest process used memory sample.

#### For mem\_tot\_avail Subevent

```
"{type %s node {%s} is_percent %s used %u avail %u diff %d sec %ld msec %ld}"
```

| Subevent Type     | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |
|-------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>type</b>       | Type of wdsysmon subevent.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |
| <b>node</b>       | Name of the node for which the total available memory is being monitored.                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |
| <b>is_percent</b> | Can be either TRUE or FALSE. TRUE means that the value is a percentage value; FALSE means that the value is an absolute value (may be an averaged value).                                                                                                                                                                                                                                                                                                                                                                                        |
| <b>used</b>       | If the <b>sec</b> and <b>msec</b> variables are specified as 0 or are unspecified in the event registration Tcl command extension, the total used memory is in the latest sample. If a time window is specified and is greater than zero in the event registration Tcl command extension, the averaged total used memory utilization is in the given time window.                                                                                                                                                                                |
| <b>avail</b>      | If the <b>sec</b> and <b>msec</b> variables are specified as 0 or are unspecified in the event registration Tcl command extension, the <b>avail</b> is in the latest total available memory sample. If a time window is specified and is greater than zero in the event registration Tcl command extension, the <b>avail</b> is the total available memory utilization in the specified time window.                                                                                                                                             |
| <b>diff</b>       | If the <b>sec</b> and <b>msec</b> variables are specified as 0 or are unspecified in the event registration Tcl command extension, the <b>diff</b> is the percentage difference between the first total available memory sample ever collected and the latest total available memory sample. If a time window is specified and is greater than zero in the event registration Tcl command extension, the <b>diff</b> is the percentage difference between the oldest and latest total available memory utilization in the specified time window. |
| <b>sec msec</b>   | If the <b>sec</b> and <b>msec</b> variables are specified as 0 or are unspecified in the event registration Tcl command extension, they are both 0. If a time window is specified and is greater than zero in the event registration Tcl command extension, they are the actual time difference between the time stamps of the oldest and latest samples in this time window.                                                                                                                                                                    |

If the **is\_percent** argument is FALSE, and the **sec** and **msec** arguments are specified as 0 or are unspecified in the event registration Tcl command extension:

- **used** is the total used memory in the latest sample.
- **avail** is the total available memory in the latest sample.
- **diff** is 0.

- **sec** and **msec** are both 0.

If the **is\_percent** argument is FALSE, and a time window is specified as greater than zero in the event registration Tcl command extension:

- **used** is 0.
- **avail** is the averaged total available memory sample value in the specified time window.
- **diff** is 0.
- **sec** and **msec** are both the actual time difference between the time stamps of the oldest and latest total available memory samples in this time window.

If the **is\_percent** argument is TRUE, and a time window is specified as greater than zero in the event registration Tcl command extension:

- **used** is 0.
- **avail** is 0.
- **diff** is the percentage difference between the oldest and latest total available memory samples in the specified time window.
- **sec** and **msec** are both the actual time difference between the time stamps of the oldest and latest total available memory samples in this time window.

If the **is\_percent** argument is TRUE, and the **sec** and **msec** arguments are specified as 0 or are unspecified in the event registration Tcl command extension:

- **used** is 0.
- **avail** is 0.
- **diff** is the percentage difference between the first total available memory sample ever collected and the latest total available memory sample.
- **sec** and **msec** are the actual time difference between the time stamps of the first total available memory sample ever collected and the latest total available memory sample.

**For mem\_tot\_used Subevent**

```
"{type %s node %s} is_percent %s used %u avail %u diff %d sec %ld msec %ld}"
```

| Subevent Type     | Description                                                                                                                                               |
|-------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>type</b>       | Type of wdsysmon subevent.                                                                                                                                |
| <b>node</b>       | Name of the node for which the total used memory is being monitored.                                                                                      |
| <b>is_percent</b> | Can be either TRUE or FALSE. TRUE means that the value is a percentage value; FALSE means that the value is an absolute value (may be an averaged value). |

| Subevent Type   | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |
|-----------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>used</b>     | If the <b>sec</b> and <b>msec</b> variables are specified as 0 or are unspecified in the event registration Tcl command extension, the total used memory is in the latest sample. If a time window is specified and is greater than zero in the event registration Tcl command extension, the averaged total used memory utilization is in the given time window.                                                                                                                                                                 |
| <b>avail</b>    | If the <b>sec</b> and <b>msec</b> variables are specified as 0 or are unspecified in the event registration Tcl command extension, the <b>avail</b> is in the latest total used memory sample. If a time window is specified and is greater than zero in the event registration Tcl command extension, the <b>avail</b> is the total used memory utilization in the specified time window.                                                                                                                                        |
| <b>diff</b>     | If the <b>sec</b> and <b>msec</b> variables are specified as 0 or are unspecified in the event registration Tcl command extension, the <b>diff</b> is the percentage difference between the first total used memory sample ever collected and the latest total used memory sample. If a time window is specified and is greater than zero in the event registration Tcl command extension, the <b>diff</b> is the percentage difference between the oldest and latest total used memory utilization in the specified time window. |
| <b>sec msec</b> | If the <b>sec</b> and <b>msec</b> variables are specified as 0 or are unspecified in the event registration Tcl command extension, they are both 0. If a time window is specified and is greater than zero in the event registration Tcl command extension, the <b>sec</b> and <b>msec</b> variables are the actual time difference between the time stamps of the oldest and latest samples in this time window.                                                                                                                 |

If the **is\_percent** argument is FALSE, and the **sec** and **msec** arguments are specified as 0 or are unspecified in the event registration Tcl command extension:

- **used** is the total used memory in the latest sample,
- **avail** is the total available memory in the latest sample,
- **diff** is 0,
- **sec** and **msec** are both 0,

If the **is\_percent** argument is FALSE, and a time window is specified as greater than zero in the event registration Tcl command extension:

- **used** is the averaged total used memory sample value in the specified time window,
- **avail** is 0,
- **diff** is 0,
- **sec** and **msec** are both the actual time difference between the time stamps of the oldest and latest total used memory samples in this time window,

If the **is\_percent** argument is TRUE, and a time window is specified as greater than zero in the event registration Tcl command extension:

- **used** is 0.
- **avail** is 0.

- **diff** is the percentage difference between the oldest and latest total used memory samples in the specified time window.
- **sec** and **msec** are both the actual time difference between the time stamps of the oldest and latest total used memory samples in this time window.

If the **is\_percent** argument is TRUE, and the sec and msec arguments are specified as 0 or are unspecified in the event registration Tcl command extension:

- **used** is 0.
- **avail** is 0.
- **diff** is the percentage difference between the first total used memory sample ever collected and the latest total used memory sample.
- **sec** and **msec** are the actual time difference between the time stamps of the first total used memory sample ever collected and the latest total used memory sample.



---

**Note** Inside a subevent description, each argument is position independent.

---







## CHAPTER 106

# EEM Event Tcl Command Extensions

The following conventions are used for the syntax documented on the Tcl command extension pages:

- An optional argument is shown within square brackets, for example:

[type ?]

- A question mark ? represents a variable to be entered.
- Choices between arguments are represented by pipes, for example:

priority low|normal|high



---

**Note** For all EEM Tcl command extensions, if there is an error, the returned Tcl result string contains the error information.

---



---

**Note** Arguments for which no numeric range is specified take an integer from -2147483648 to 2147483647, inclusive.

---

- [event\\_completion](#), on page 2263
- [event\\_completion\\_with\\_wait](#), on page 2264
- [event\\_publish](#), on page 2265
- [event\\_wait](#), on page 2268

## event\_completion

Sends a notification to the EEM server that the policy is done servicing the event that triggered it. The event only takes a single argument which is the **return\_code** of this event instance.

### Syntax

```
event_completion status ?
```

**Arguments**

|        |                                                                                                                                                  |
|--------|--------------------------------------------------------------------------------------------------------------------------------------------------|
| status | (Mandatory) Exit status (return_code) of this event instance. A value of zero indicates no error and any other integer value indicates an error. |
|--------|--------------------------------------------------------------------------------------------------------------------------------------------------|

**Result String**

None

**Set\_cerrno**

No

## event\_completion\_with\_wait

The **event\_completion\_with\_wait** command combines the two commands **event\_completion** and **event\_wait** into a single command for ease of use.

The **event\_completion** command sends a notification to the EEM server that the policy is done servicing the event that triggered it. The event only takes a single argument which is the **return\_code** of this event instance.

The **event\_wait** places the Tcl policy into a sleep state. When the Tcl policy receives a new signal announcing a new event, the policy is placed into a wake state and again returns to a sleep state. This loop continues. If **event\_wait** policy is invoked before **event\_completed** policy, an error results and the policy exits.

**Syntax**

```
event_completion_with_wait status ? [refresh_vars]
```

**Arguments**

|              |                                                                                                                                                        |
|--------------|--------------------------------------------------------------------------------------------------------------------------------------------------------|
| status       | (Mandatory) exit_status (return_code) of this event instance. A value of zero indicates no error. Any other integer value indicates an error.          |
| refresh_vars | (Optional) Indicates whether built-in and environment variables should be updated (refreshed) from the EEM Policy Director during this event instance. |

**Result String**

None

**Set\_cerrno**

Yes

**Sample Usage**

Here is a similar example as above using this single command:

```
namespace import ::cisco::eem::*
namespace import ::cisco::lib::*
```

```

set i 1
while {1 == 1} { # Start high performance policy loop
 array set arr_einfo [event_reqinfo]
 if {$_cerno != 0} {
 set result [format "component=%s; subsys err=%s; posix err=%s;\n%s" \
 $_cerr_sub_num $_cerr_sub_err $_cerr_posix_err $_cerr_str]
 error $result
 }
 action_syslog msg "event $i serviced" priority info
 if {$i == 5} {
 action_syslog msg "Exiting after servicing 5 events" priority info
 exit 0
 }
 incr i
 array set _event_state_arr [event_completion_with_wait status 0 refresh_vars 1]
 if {$_event_state_arr(event_state) != 0} {
 action_syslog msg "Exiting: failed event_state " \
 "$event_state_arr(event_state)" priority info
 exit 0
 }
}
}

```



**Note** The running configuration output is the same as the event\_publishTcl command.

## event\_publish

Publishes an application-specific event.

### Syntax

```
event_publish sub_system ? type ? [arg1 ?] [arg2 ?] [arg3 ?] [arg4 ?]
```

### Arguments

|                   |                                                                                                                                                                                             |
|-------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| sub_system        | (Mandatory) Number assigned to the EEM policy that published the application-specific event. Number is set to 798 because all other numbers are reserved for Cisco use.                     |
| type              | (Mandatory) Event subtype within the specified component. The sub_system and type arguments uniquely identify an application event. Must be an integer between 1 and 4294967295, inclusive. |
| [arg1 ?]-[arg4 ?] | (Optional) Four pieces of application event publisher string data.                                                                                                                          |

### Result String

None

### Set\_cerno

Yes

```
(_cerr_sub_err = 2) FH_ESYSERR (generic/unknown error from OS/system)
```

This error means that the operating system reported an error. The POSIX errno value that is reported with the error should be used to determine the cause of the operating system error.

### Sample Usage

This example demonstrates how to use the **event\_publish** Tcl command extension to execute a script *n* times repeatedly to perform some function (for example, to measure the amount of CPU time taken by a given group of Tcl statements). This example uses two Tcl scripts.

Script1 publishes a type 9999 EEM event to cause Script2 to run for the first time. Script1 is registered as a none event and is run using the Cisco IOS CLI **event manager run** command. Script2 is registered as an EEM application event of type 9999, and this script checks to see if the application publish arg1 data (the iteration number) exceeds the EEM environment variable test\_iterations value. If the test\_iterations value is exceeded, the script writes a message and exits; otherwise the script executes the remaining statements and reschedules another run. To measure the CPU utilization for Script2, use a value of test\_iterations that is a multiple of 10 to calculate the amount of average CPU time used by Script2.

To run the Tcl scripts, enter the following Cisco IOS commands:

```
configure terminal
 event manager environment test_iterations 100
 event manager policy script1.tcl
 event manager policy script2.tcl
 end
 event manager run script1.tcl
```

The Tcl script Script2 will be executed 100 times. If you execute the script without the extra processing and derive the average CPU utilization, and then add the extra processing and repeat the test, you can subtract the former CPU utilization from the later CPU utilization to determine the average for the extra processing.

#### Script1 (script1.tcl)

```
::cisco::eem::event_register_none
namespace import ::cisco::eem::*
namespace import ::cisco::lib::*
Query the event info.
array set arr_einfo [event_reqinfo]
if {$_cerrno != 0} {
 set result [format "component=%s; subsys err=%s; posix err=%s;\n%s" \
 $_cerr_sub_num $_cerr_sub_err $_cerr_posix_err $_cerr_str]
 error $result
}

action_syslog priority info msg "EEM application_publish test start"
if {$_cerrno != 0} {
 set result [format \
 "component=%s; subsys err=%s; posix err=%s;\n%s" \
 $_cerr_sub_num $_cerr_sub_err $_cerr_posix_err $_cerr_str]
 error $result
}

Cause the first iteration to run.
event_publish sub_system 798 type 9999 arg1 0
if {$_cerrno != 0} {
 set result [format \
 "component=%s; subsys err=%s; posix err=%s;\n%s" \
 $_cerr_sub_num $_cerr_sub_err $_cerr_posix_err $_cerr_str]
```

```

 error $result
}

Script2 (script2.tcl)

::cisco::eem::event_register_appl sub_system 798 type 9999

Check if all the required environment variables exist.
If any required environment variable does not exist, print out an error msg and quit.
if (![info exists test_iterations]) {
 set result \
 "Policy cannot be run: variable test_iterations has not been set"
 error $result $errorInfo
}

namespace import ::cisco::eem::*
namespace import ::cisco::lib::*

Query the event info.
array set arr_einfo [event_reqinfo]
if {$_cerrno != 0} {
 set result [format "component=%s; subsys err=%s; posix err=%s;\n%s" \
 $_cerr_sub_num $_cerr_sub_err $_cerr_posix_err $_cerr_str]
 error $result
}

Data1 contains the arg1 value used to publish this event.
set iter $arr_einfo(data1)

Use the arg1 info from the previous run to determine when to end.
if {$iter >= $test_iterations} {
 # Log a message.
 action_syslog priority info msg "EEM application_publish test end"
 if {$_cerrno != 0} {
 set result [format \
 "component=%s; subsys err=%s; posix err=%s;\n%s" \
 $_cerr_sub_num $_cerr_sub_err $_cerr_posix_err $_cerr_str]
 error $result
 }
 exit 0
}
set iter [expr $iter + 1]

Log a message.
set msg [format "EEM application_publish test iteration %s" $iter]
action_syslog priority info msg $msg
if {$_cerrno != 0} {
 set result [format "component=%s; subsys err=%s; posix err=%s;\n%s" \
 $_cerr_sub_num $_cerr_sub_err $_cerr_posix_err $_cerr_str]
 error $result
}

Do whatever processing that you want to measure here.

Cause the next iteration to run. Note that the iteration is passed to the
next operation as arg1.
event_publish sub_system 798 type 9999 arg1 $iter
if {$_cerrno != 0} {
 set result [format \
 "component=%s; subsys err=%s; posix err=%s;\n%s" \
 $_cerr_sub_num $_cerr_sub_err $_cerr_posix_err $_cerr_str]
 error $result
}
}

```

## event\_wait

Places the Tcl policy into a sleep state. When the Tcl policy receives a new signal announcing a new event, the policy is placed into a wake state and again returns to a sleep state. This loop continues. If **event\_wait** policy is invoked before **event\_completed** policy, an error results and the policy exits.

### Syntax

```
event_wait [refresh_vars]
```

### Arguments

|              |                                                                                                                                                        |
|--------------|--------------------------------------------------------------------------------------------------------------------------------------------------------|
| refresh_vars | (Optional) Indicates whether built-in and environment variables should be updated (refreshed) from the EEM Policy Director during this event instance. |
|--------------|--------------------------------------------------------------------------------------------------------------------------------------------------------|

### Result String

None

### Set\_cerrno

No

### Sample Usage

The **event\_wait** event detector returns an array type value with a single element named **event\_state**. Event\_state is a value sent back from the EEM Server indicating whether or not an error has occurred in processing the event. An example of an error here would be if the user configured **event\_wait** before configuring **event\_completion** when handling the event instance.

The following sample output shows the use of both **event\_completion** and **event\_wait**Tcl commands:

```
::cisco::eem::event_register_syslog tag e1 occurs 1 pattern CLEAR maxrun 0
namespace import ::cisco::eem::*
namespace import ::cisco::lib::*
set i 1
while {1 == 1} { # Start high performance policy loop
 array set arr_einfo [event_reqinfo]
 if {$_cerrno != 0} {
 set result [format "component=%s; subsys err=%s; posix err=%s;\n%s" \
 $_cerr_sub_num $_cerr_sub_err $_cerr_posix_err $_cerr_str]
 error $result
 }
 action_syslog msg "event $i serviced" priority info
 if {$i == 5} {
 action_syslog msg "Exiting after servicing 5 events" priority info
 exit 0
 }
 incr i
 event_completion status 0
 array set _event_state_arr [event_wait refresh_vars 0]
 if {$_event_state_arr(event_state) != 0} {
 action_syslog msg "Exiting: failed event_state " \
 " $_event_state_arr(event_state)" priority info
 }
}
```

```

 exit 0
 }
}

```

Here is an example of the running configuration:

```

Device#
01:00:44: %SYS-5-CONFIG_I: Configured from console by consoleclear counters
Clear "show interface" counters on all interfaces [confirm]
Device#
01:00:49: %CLEAR-5-COUNTERS: Clear counter on all interfaces by console
01:00:49: %HA_EM-6-LOG: high_perf_example.tcl: event 1 serviced
Device#
Device#clear counters
Clear "show interface" counters on all interfaces [confirm]
Device#
Device#
01:00:53: %CLEAR-5-COUNTERS: Clear counter on all interfaces by console
01:00:53: %HA_EM-6-LOG: high_perf_example.tcl: event 2 serviced
Device#clear counters
Clear "show interface" counters on all interfaces [confirm]
Device#
Device#
01:00:56: %CLEAR-5-COUNTERS: Clear counter on all interfaces by console
01:00:56: %HA_EM-6-LOG: high_perf_example.tcl: event 3 serviced
Device#
Device#
Device#clear counters
Clear "show interface" counters on all interfaces [confirm]
Device#
01:00:59: %CLEAR-5-COUNTERS: Clear counter on all interfaces by console
Device#
01:00:59: %HA_EM-6-LOG: high_perf_example.tcl: event 4 serviced
01:00:59: %HA_EM-6-LOG: high_perf_example.tcl: Exiting after servicing 5 events
Device#
Device#
Device#copy tftp disk1:
Address or name of remote host [dirt]?
Source filename [user/eem_scripts/high_perf_example.tcl]?
Destination filename [high_perf_example.tcl]?
%Warning:There is a file already existing with this name
Do you want to over write? [confirm]
Accessing tftp://dirt/user/eem_scripts/high_perf_example.tcl...
Loading user/eem_scripts/high_perf_example.tcl from 192.0.2.19 (via FastEthernet0/0): !
[OK - 909 bytes]
909 bytes copied in 0.360 secs (2525 bytes/sec)
Device#
Device#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Device(config)#no event manager policy high_perf_example.tcl
Device(config)#event manager po high_perf_example.tcl
Device(config)#end
Device#
Device#
Device#
Device#
01:02:19: %SYS-5-CONFIG_I: Configured from console by consoleclear counters
Clear "show interface" counters on all interfaces [confirm]
Device#
01:02:23: %CLEAR-5-COUNTERS: Clear counter on all interfaces by console
Device#
Device#
01:02:23: %HA_EM-6-LOG: high_perf_example.tcl: event 1 serviced
Device#

```

```

Device#clear counters
Clear "show interface" counters on all interfaces [confirm]
Device#
Device#
01:02:26: %CLEAR-5-COUNTERS: Clear counter on all interfaces by console
01:02:26: %HA_EM-6-LOG: high_perf_example.tcl: event 2 serviced
Device#
Device#clear counters
Clear "show interface" counters on all interfaces [confirm]
Device#
Device#
01:02:29: %CLEAR-5-COUNTERS: Clear counter on all interfaces by console
01:02:29: %HA_EM-6-LOG: high_perf_example.tcl: event 3 serviced
Device#
Device#clear counters
Clear "show interface" counters on all interfaces [confirm]
Device#
Device#
01:02:33: %CLEAR-5-COUNTERS: Clear counter on all interfaces by console
Device#
01:02:33: %HA_EM-6-LOG: high_perf_example.tcl: event 4 serviced
Device#
Device#clear counters
Clear "show interface" counters on all interfaces [confirm]
Device#
Device#
Device#
01:02:36: %CLEAR-5-COUNTERS: Clear counter on all interfaces by console
01:02:36: %HA_EM-6-LOG: high_perf_example.tcl: event 5 serviced
01:02:36: %HA_EM-6-LOG: high_perf_example.tcl: Exiting after servicing 5 events
Device#

```

Also while an event has been serviced and is waiting for the next event to come in **show event manager policy active** command will display the following output:

```

Device#show event manager policy active
Key: p - Priority :L - Low, H - High, N - Normal, Z - Last
 s - Scheduling node :A - Active, S - Standby
default class - 1 script event
no. job id p s status time of event event type name
1 11 N A wait Mon Oct20 14:15:24 2008 syslog
high_perf_example.tcl

```

In the above example the status is wait. This indicates that the policy is waiting for the next event to come in.





## CHAPTER 107

# EEM Library Debug Command Extensions

- [cli\\_debug](#), on page 2271
- [smtp\\_debug](#), on page 2271

## cli\_debug

Prints a command-line interface (CLI) debug statement to syslog. This Tcl command extension is used to print a CLI debug statement to syslog if the **debug event manager tcl cli\_library** Cisco IOS CLI command is in effect.

### Syntax

```
cli_debug spec_string debug_string
```

### Arguments

|              |                                                                                       |
|--------------|---------------------------------------------------------------------------------------|
| spec_string  | (Mandatory) The spec_string argument is used to indicate the type of debug statement. |
| debug_string | (Mandatory) The debug_string argument is used to indicate the debugging text.         |

### Result String

None

### Set\_cerrno

No

## smtp\_debug

Prints a Simple Mail Transfer Protocol (SMTP) debug statement to syslog. This Tcl command extension prints a SMTP debug statement to syslog if the **debug event manager tcl smtp\_library** Cisco IOS command-line interface (CLI) command is in effect.

## Syntax

```
smtp_debug spec_string debug_string
```

## Arguments

|              |                                                                                       |
|--------------|---------------------------------------------------------------------------------------|
| spec_string  | (Mandatory) The spec_string argument is used to indicate the type of debug statement. |
| debug_string | (Mandatory) The debug_string argument is used to indicate the debugging text.         |

## Result String

None

## Set\_cerrno

No



## CHAPTER 108

# EEM Multiple Event Support Tcl Command Extensions

---

The following conventions are used for the syntax documented on the Tcl command extension pages:

- An optional argument is shown within square brackets, for example:

[type ?]

- A question mark ? represents a variable to be entered.
- Choices between arguments are represented by pipes, for example:

priority low|normal|high



---

**Note** For all EEM Tcl command extensions, if there is an error, the returned Tcl result string contains the error information.

---



---

**Note** Arguments for which no numeric range is specified take an integer from -2147483648 to 2147483647, inclusive.

---

- [attribute](#), on page 2273
- [correlate](#), on page 2274
- [trigger](#), on page 2275

## attribute

Specifies a complex event.

### Syntax

```
attribute tag ? [occurs ?]
```

**Arguments**

|               |                                                                                                                                                                                          |
|---------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>tag</b>    | Specifies a tag using the <i>event-tag</i> argument that can be used with the <b>attribute</b> command to associate an event.                                                            |
| <b>occurs</b> | (Optional) Specifies the number of occurrences before an EEM event is triggered. If not specified, an EEM event is triggered on the first occurrence. The range is from 1 to 4294967295. |

**Result String**

None

**Set\_cerrno**

No

# correlate

Builds a single complex event and allows boolean logic to relate events and tracked objects.

**Syntax**

```
correlate event ? track ? [andnot | and | or] event ? track ?
```

**Arguments**

|               |                                                                                                                                                                                                                                                                                                                        |
|---------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>event</b>  | Specifies the event that can be used with the <b>trigger</b> command to support multiple event statements within an script.<br><br>If the event associated with the <i>event-tag</i> argument occurs for the number of times specified by the <b>trigger</b> command, the result is true. If not, the result is false. |
| <b>track</b>  | Specifies the event object number for tracking. The range is from 1 to 500.<br><br>If the tracked object is set, the result of the evaluation is true. If the tracked object is not set or is undefined, the result of the evaluation is false. This result is regardless of the state of the object.                  |
| <i>andnot</i> | (Optional) Specifies that if event 1 occurs the action is executed, and if event 2 and event 3 occur together the action is not executed.                                                                                                                                                                              |
| and           | (Optional) Specifies that if event 1 occurs the action is executed, and if event 2 and event 3 occur together the action is executed.<br><br><b>Note</b> When "and" is used to group events such as traps or syslog messages, then the default trigger occurrence window is three minutes.                             |
| or            | (Optional) Specifies that if event 1 occurs the action is executed, or else if event 2 and event 3 occur together the action is executed.                                                                                                                                                                              |

**Result String**

None

**Set \_cerno**

No

# trigger

Specifies the multiple event configuration ability of Embedded Event Manager (EEM) events. A multiple event is one that can involve one or more event occurrences, one or more tracked object states, and a time period for the event to occur. The events are raised based on the specified parameters.

**Syntax**

```
trigger [occurs ?] [period ?] [period-start ?] [delay ?]
```

**Arguments**

|                     |                                                                                                                                                                                                                                                                                                                                                                     |
|---------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>occurs</b>       | (Optional) Specifies the number of times the total correlation occurs before an EEM event is raised. When a number is not specified, an EEM event is raised on the first occurrence. The range is from 1 to 4294967295.                                                                                                                                             |
| <b>period</b>       | (Optional) Time interval in seconds and optional milliseconds, during which the one or more occurrences must take place. This is specified in the format ssssssss[.mmm], where ssssssss must be an integer number representing seconds between 0 and 4294967295, inclusive and mmm represents milliseconds and must be an integer number between 0 to 999.          |
| <b>period-start</b> | (Optional) Specifies the start of an event correlation window. If not specified, event monitoring is enabled after the first CRON period occurs.                                                                                                                                                                                                                    |
| <b>delay</b>        | (Optional) Specifies the number of seconds and optional milliseconds after which an event will be raised if all the conditions are true (specified in the format ssssssss[.mmm], where ssssssss must be an integer number representing seconds between 0 and 4294967295, inclusive and mmm represents milliseconds and must be an integer number between 0 to 999). |

**Result String**

None

**Set \_cerno**

No

trigger



## CHAPTER 109

# EEM SMTP Library Command Extensions

All Simple Mail Transfer Protocol (SMTP) library command extensions belong to the `::cisco::lib` namespace.

To use this library, the user needs to provide an e-mail template file. The template file can include Tcl global variables so that the e-mail service and the e-mail text can be configured through the **event manager environment Cisco IOS** command-line interface (CLI) configuration command. There are commands in this library to substitute the global variables in the e-mail template file and to send the desired e-mail context with the To address, CC address, From address, and Subject line properly configured using the configured e-mail server.

### E-Mail Template

The e-mail template file has the following format:



---

**Note** Based on RFC 2554, the SMTP e-mail server name--Mailservername-- can be in any one of the following template formats: `username:password@host`, `username@host`, or `host`.

---

```
Mailservername:<space><the list of candidate SMTP server addresses>
From:<space><the e-mail address of sender>
To:<space><the list of e-mail addresses of recipients>
Cc:<space><the list of e-mail addresses that the e-mail will be copied to>
Sourceaddr:<space><the IP addresses of the recipients>
Subject:<subject line>
<a blank line>
<body>
```



---

**Note** Note that the template normally includes Tcl global variables for configuration.

---

In a Tcl policy, the port number can be specified by a "Port" line in the e-mail template. If port is not specified, the default port of 25 is used.

Below is a sample e-mail template file:

```
Mailservername: $_email_server
From: $_email_from
To: $_email_to
Cc: $_email_cc
```

```
Sourceaddr: $_email_ipaddr
Port: <port number>
Subject: From router $routername: Process terminated
process name: $process_name
subsystem: $sub_system
exit status: $exit_status
respawn count: $respawn_count
```

- [smtp\\_send\\_email, on page 2278](#)
- [smtp\\_subst, on page 2279](#)

## smtp\_send\_email

Given the text of an e-mail template file with all global variables already substituted, sends the e-mail out using Simple Mail Transfer Protocol (SMTP). The e-mail template specifies the candidate mail server addresses, To addresses, CC addresses, From address, subject line, and e-mail body.




---

**Note** A list of candidate e-mail servers can be provided so that the library will try to connect the servers on the list one by one until it can successfully connect to one of them.

---

### Syntax

```
smtp_send_email text
```

### Arguments

<code>text</code>	(Mandatory) The text of an e-mail template file with all global variables already substituted.
-------------------	------------------------------------------------------------------------------------------------

### Result String

None

### Set \_cerrno

- Wrong 1st line format--Mailservername:list of server names.
- Wrong 2nd line format--From:from-address.
- Wrong 3rd line format--To:list of to-addresses.
- Wrong 4th line format--CC:list of cc-addresses.
- Error connecting to mail server:--\$sock closed by remote server (where \$sock is the name of the socket opened to the mail server).
- Error connecting to mail server:--\$sock reply code is \$k instead of the service ready greeting (where \$sock is the name of the socket opened to the mail server; \$k is the reply code of \$sock).
- Error connecting to mail server:--cannot connect to all the candidate mail servers.
- Error disconnecting from mail server:--\$sock closed by remote server (where \$sock is the name of the socket opened to the mail server).



## Sample Scripts

After all needed global variables in the e-mail template are defined:

```
if [catch {smtp_subst [file join $tcl_library email_template_sm]} result] {
 puts stderr $result
 exit 1
}
if [catch {smtp_send_email $result} result] {
 puts stderr $result
 exit 1
}
```

# smtp\_subst

Given an e-mail template file `e-mail_template`, substitutes each global variable in the file by its user-defined value. Returns the text of the file after substitution.

## Syntax

```
smtp_subst e-mail_template
```

## Arguments

e-mail_template	(Mandatory) Name of an e-mail template file in which global variables need to be substituted by a user-defined value. An example filename could be <code>/disk0://example.template</code> which represents a file named <code>example.template</code> in a top-level directory on an ATA flash disk in slot 0.
-----------------	----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

## Result String

The text of the e-mail template file with all the global variables substituted.

## Set \_cerrno

- cannot open e-mail template file
- cannot close e-mail template file

smtp\_subst



## CHAPTER 110

# EEM System Information Tcl Command Extensions

---

The following conventions are used for the syntax documented on the Tcl command extension pages:

- An optional argument is shown within square brackets, for example:

[type ?]

- A question mark ? represents a variable to be entered.
- Choices between arguments are represented by pipes, for example:

priority low|normal|high



---

**Note** All EEM system information commands--`sys_reqinfo_XXX`--have the Set \_cerno section set to yes.

---



---

**Note** For all EEM Tcl command extensions, if there is an error, the returned Tcl result string contains the error information.

---



---

**Note** Arguments for which no numeric range is specified take an integer from -2147483648 to 2147483647, inclusive.

---

- [sys\\_reqinfo\\_cli\\_freq](#), on page 2282
- [sys\\_reqinfo\\_cli\\_history](#), on page 2283
- [sys\\_reqinfo\\_cpu\\_all](#), on page 2283
- [sys\\_reqinfo\\_crash\\_history](#), on page 2284
- [sys\\_reqinfo\\_mem\\_all](#), on page 2285
- [sys\\_reqinfo\\_proc](#), on page 2286
- [sys\\_reqinfo\\_proc\\_all](#), on page 2288
- [sys\\_reqinfo\\_routename](#), on page 2288
- [sys\\_reqinfo\\_snmp](#), on page 2289
- [sys\\_reqinfo\\_syslog\\_freq](#), on page 2290

- [sys\\_reqinfo\\_syslog\\_history](#), on page 2291

## sys\_reqinfo\_cli\_freq

Queries the frequency information of all command-line interface (CLI) events.

### Syntax

```
sys_reqinfo_cli_freq
```

### Arguments

None

### Result String

```
rec_list {{CLI frequency string 0},{CLI frequency str 1}, ...}
```

Where each CLI frequency string is:

```
time_sec %ld time_msec %ld match_count %u raise_count %u occurs %u period_sec %ld period_msec %ld
pattern {%s}
```

rec_list	Marks the start of the CLI event frequency list.
time_sec time_msec	Last time when this CLI event was raised.
match count	Number of times that a CLI command matches the pattern specified by this CLI event specification.
raise_count	Number of times that this CLI event was raised. The following fields are information about the CLI event specification: <ul style="list-style-type: none"> <li>• sync--A "yes" means that event publish should be performed synchronously. The event detector will be notified when the Event Manager Server has completed publishing the event. The Event Manager Server will return a code that indicates whether or not the CLI command should be executed.</li> <li>• skip--A "yes" means that the CLI command should not be executed if the sync flag is not set.</li> </ul>
occurs	Number of occurrences before an event is raised; if this argument is not specified, an event is raised on the first occurrence.
period_sec period_msec	Number of occurrences must occur within this number of POSIX timer units in order to raise event; if this argument is not specified, it does not apply.
pattern	Regular expression used to perform CLI command pattern matching.

### Set\_cerrno

Yes

## sys\_reqinfo\_cli\_history

Queries the history of command-line interface (CLI) commands.

### Syntax

```
sys_reqinfo_cli_history
```

### Arguments

None

### Result String

```
rec_list {{CLI history string 0}, {CLI history str 1},...}
```

Where each CLI history string is:

```
time_sec %ld time_msec %ld cmd {%s}
```

rec_list	Marks the start of the CLI command history list.
time_sec time_msec	Time when the CLI command was run.
cmd	Text of the CLI command.

### Set\_cerrno

Yes

## sys\_reqinfo\_cpu\_all

Queries the CPU utilization of the top processes (both POSIX processes and IOS processes) during a specified time period and in a specified order. This Tcl command extension is supported only in Software Modularity images.

### Syntax

```
sys_reqinfo_cpu_all order cpu_used [sec ?] [msec ?] [num ?]
```

### Arguments

order	(Mandatory) Order used for sorting the CPU utilization of processes.
cpu_used	(Mandatory) Specifies that the average CPU utilization, for the specified time window, will be sorted in descending order.

sec msec	(Optional) The time period, in seconds and milliseconds, during which the average CPU utilization is calculated. Must be integers in the range from 0 to 4294967295. If not specified, or if both sec and msec are specified as 0, the most recent CPU sample is used.
num	(Optional) Number of entries from the top of the sorted list of processes to be displayed. Must be an integer in the range from 1 to 4294967295. Default value is 5.

**Result String**

```
rec_list {{process CPU info string 0},{process CPU info string 1}, ...}
```

Where each process CPU info string is:

```
pid %u name {%s} cpu_used %u
```

rec_list	Marks the start of the process CPU information list.
pid	Process ID.
name	Process name.
cpu_used	Specifies that if sec and msec are specified with a number greater than zero, the average percentage is calculated from the process CPU utilization during the specified time period. If sec and msec are both zero or not specified, the average percentage is calculated from the process CPU utilization in the latest sample.

**Set\_cerrno**

Yes

## sys\_reqinfo\_crash\_history

Queries the crash information of all processes that have ever crashed. This Tcl command extension is supported only in Software Modularity images.

**Syntax**

```
sys_reqinfo_crash_history
```

**Arguments**

None

**Result String**

```
rec_list {{crash info string 0},{crash info string 1}, ...}
```

Where each crash info string is:

```
job_id %u name {%s} respawn_count %u fail_count %u dump_count %u
inst_id %d exit_status 0x%x exit_type %d proc_state {%s} component_id 0x%x
crash_time_sec %ld crash_time_msec %ld
```

job_id	System manager assigned job ID for the process. An integer between 1 and 4294967295, inclusive.
name	Process name.
respawn_count	Total number of restarts for the process.
fail_count	Number of restart attempts of the process. This count is reset to zero when the process is successfully restarted.
dump_count	Number of core dumps performed.
inst_id	Process instance ID.
exit_status	Last exit status of the process.
exit_type	Last exit type.
proc_state	Sysmgr process states. One of the following: error, forced_stop, hold, init, ready_to_run, run, run_rnode, stop, waitEOltimer, wait_rnode, wait_spawntimer, wait_tpl.
component_id	Version manager assigned component ID for the component to which the process belongs.
crash_time_sec crash_time_msec	Seconds and milliseconds since January 1, 1970, which represent the last time the process crashed.

**Set \_cerrno**

Yes

## sys\_reqinfo\_mem\_all

Queries the memory usage of the top processes (both POSIX and IOS) during a specified time period and in a specified order. This Tcl command extension is supported only in Software Modularity images.

**Syntax**

```
sys_reqinfo_mem_all order allocates|increase|used [sec ?] [msec ?] [num ?]
```

**Arguments**

order	(Mandatory) Order used for sorting the memory usage of processes.
allocates	(Mandatory) Specifies that the memory usage is sorted by the number of process allocations during the specified time window, and in descending order.
increase	(Mandatory) Specifies that the memory usage is sorted by the percentage of process memory increase during the specified time window, and in descending order.
used	(Mandatory) Specifies that the memory usage is sorted by the current memory used by the process.

sec msec	(Optional) The time period, in seconds and milliseconds, during which the process memory usage is calculated. Must be integers in the range from 0 to 4294967295. If both sec and msec are specified and are nonzero, the number of allocations is the difference between the number of allocations in the oldest and latest samples collected in the time period. The percentage is calculated as the percentage difference between the memory used in the oldest and latest samples collected in the time period. If not specified, or if both sec and msec are specified as 0, the first sample ever collected is used as the oldest sample; that is, the time period is set to be the time from startup until the current moment.
num	(Optional) Number of entries from the top of the sorted list of processes to be displayed. Must be an integer in the range from 1 to 4294967295. Default value is 5.

### Result String

```
rec_list {{process mem info string 0},{process mem info string 1}, ...}
```

Where each process mem info string is:

```
pid %u name {%s} delta_allocs %d initial_alloc %u current_alloc %u percent_increase %d
```

rec_list	Marks the start of the process memory usage information list.
pid	Process ID.
name	Process name.
delta_allocs	Specifies the difference between the number of allocations in the oldest and latest samples collected in the time period.
initial_alloc	Specifies the amount of memory, in kilobytes, used by the process at the start of the time period.
current_alloc	Specifies the amount of memory, in kilobytes, currently used by the process.
percent_increase	Specifies the percentage difference between the memory used in the oldest and latest samples collected in the time period. The percentage difference can be expressed as $\text{current\_alloc} - \text{initial\_alloc} \times 100$ and divided by $\text{initial\_alloc}$ .

### Set\_cerrno

Yes

## sys\_reqinfo\_proc

Queries the information about a single POSIX process. This Tcl command extension is supported only in Software Modularity images.

### Syntax

```
sys_reqinfo_proc job_id ?
```



**Arguments**

job_id	(Mandatory) System manager assigned job ID for the process. Must be an integer between 1 and 4294967295, inclusive.
--------	---------------------------------------------------------------------------------------------------------------------

**Result String**

```
job_id %u component_id 0x%x name {%s} helper_name {%s} helper_path {%s} path {%s}
node_name {%s} is_respawn %u is_mandatory %u is_hold %u dump_option %d
max_dump_count %u respawn_count %u fail_count %u dump_count %u
last_respawn_sec %ld last_respawn_msec %ld inst_id %u proc_state %s
level %d exit_status 0x%x exit_type %d
```

job_id	System manager assigned job ID for the process. An integer between 1 and 4294967295, inclusive.
component_id	Version manager assigned component ID for the component to which the process belongs.
name	Process name.
helper_name	Helper process name.
helper_path	Executable path of the helper process.
path	Executable path of the process.
node_name	System manager assigned node name for the node to which the process belongs.
is_respawn	Flag that specifies that the process can be respawned.
is_mandatory	Flag that specifies that the process must be alive.
is_hold	Flag that specifies that the process is spawned until called by the API.
dump_option	Core dumping options.
max_dump_count	Maximum number of core dumping permitted.
respawn_count	Total number of restarts for the process.
fail_count	Number of restart attempts of the process. This count is reset to zero when the process is successfully restarted.
dump_count	Number of core dumps performed.
last_respawn_sec last_respawn_msec	Seconds and milliseconds in POSIX timer units since January 1, 1970, which represent the last time the process was started.
inst_id	Process instance ID.
proc_state	Sysmgr process states. One of the following: error, forced_stop, hold, init, ready_to_run, run, run_rnode, stop, waitEOltimer, wait_rnode, wait_spawntimer, wait_tpl.

level	Process run level.
exit_status	Last exit status of the process.
exit_type	Last exit type.

**Set\_cerrno**

Yes

## sys\_reqinfo\_proc\_all

Queries the information of all POSIX processes. This Tcl command extension is supported only in Software Modularity images.

**Syntax**

```
sys_reqinfo_proc_all
```

**Arguments**

None

**Result String**

```
rec_list {{process info string 0}, {process info string 1},...}
```

Where each process info string is the same as the result string of the **sysreq\_info\_proc** Tcl command extension.

**Set\_cerrno**

Yes

## sys\_reqinfo\_routename

Queries the device name.

**Syntax**

```
sys_reqinfo_routename
```

**Arguments**

None

**Result String**

```
routename %s
```

Where routename is the name of the device.

### Set\_cerrno

Yes

## sys\_reqinfo\_snmp

Queries the value of the entity specified by a Simple Network Management Protocol (SNMP) object ID.

### Syntax

```
sys_reqinfo_snmp oid ? get_type exact|next
```

### Arguments

oid	(Mandatory) SNMP OID in dot notation (for example, 1.3.6.1.2.1.2.1.0).
get_type	(Mandatory) Type of SNMP get operation that needs to be applied to the specified oid. If the get_type is "exact," the value of the specified oid is retrieved; if the get_type is "next," the value of the lexicographical successor to the specified oid is retrieved.

### Result String

```
oid {%s} value {%s}
```

oid	SNMP OID.
value	Value string of the associated SNMP data element.

### Set\_cerrno

Yes

```
(_cerr_sub_err = 2) FH_ESYSERR (generic/unknown error from OS/system)
```

This error means that the operating system reported an error. The POSIX errno value that is reported with the error should be used to determine the cause of the operating system error.

```
(_cerr_sub_err = 22) FH_ENULLPTR (event detector internal error - ptr is null)
```

This error means that an internal EEM event detector pointer was null when it should have contained a value.

```
(_cerr_sub_err = 37) FH_ENOSNMPDATA (can't retrieve data from SNMP)
```

This error means that there was no data for the SNMP object type.

```
(_cerr_sub_err = 51) FH_ESTATSTYP (invalid statistics data type)
```

This error means that the SNMP statistics data type was invalid.

```
(_cerr_sub_err = 54) FH_EFDUNAVAIL (connection to event detector unavailable)
```

This error means that the event detector was unavailable.

## sys\_reqinfo\_syslog\_freq

Queries the frequency information of all syslog events.

### Syntax

```
sys_reqinfo_syslog_freq
```

### Arguments

None

### Result String

```
rec_list {{event frequency string 0}, {log freq str 1}, ...}
```

Where each event frequency string is:

```
time_sec %ld time_msec %ld match_count %u raise_count %u occurs %u
period_sec %ld period_msec %ld pattern {%s}
```

time_sec time_msec	Seconds and milliseconds in POSIX timer units since January 1, 1970, which represent the time the last event was raised.
match_count	Number of times that a syslog message matches the pattern specified by this syslog event specification since event registration.
raise_count	Number of times that this syslog event was raised.
occurs	Number of occurrences needed in order to raise the event; if not specified, the event is raised on the first occurrence.
period_sec period_msec	Number of occurrences must occur within this number of POSIX timer units in order to raise the event; if not specified, the period check does not apply.
pattern	Regular expression used to perform syslog message pattern matching.

### Set\_cerrno

Yes

```
(_cerr_sub_err = 2) FH_ESYSERR (generic/unknown error from OS/system)
```

This error means that the operating system reported an error. The POSIX errno value that is reported with the error should be used to determine the cause of the operating system error.

```
(_cerr_sub_err = 9) FH_EMEMORY (insufficient memory for request)
```

This error means that an internal EEM request for memory failed.

```
(_cerr_sub_err = 22) FH_ENULLPTR (event detector internal error - ptr is null)
```

This error means that an internal EEM event detector pointer was null when it should have contained a value.

```
(_cerr_sub_err = 45) FH_ESEQNUM (sequence or workset number out of sync)
```

This error means that the event detector sequence or workset number was invalid.

```
(_cerr_sub_err = 46) FH_EREGEMPTY (registration list is empty)
```

This error means that the event detector registration list was empty.

```
(_cerr_sub_err = 54) FH_EFDUNAVAIL (connection to event detector unavailable)
```

This error means that the event detector was unavailable.

## sys\_reqinfo\_syslog\_history

Queries the history of the specified syslog message.

### Syntax

```
sys_reqinfo_syslog_history
```

### Arguments

None

### Result String

```
rec_list {{log hist string 0}, {log hist str 1}, ...}
```

Where each log hist string is:

```
time_sec %ld time_msec %ld msg {%s}
```

time_sec time_msec	Seconds and milliseconds since January 1, 1970, which represent the time the message was logged.
msg	Syslog message.

### Set\_cerrno

Yes

```
(_cerr_sub_err = 2) FH_ESYSERR (generic/unknown error from OS/system)
```

This error means that the operating system reported an error. The POSIX errno value that is reported with the error should be used to determine the cause of the operating system error.

```
(_cerr_sub_err = 22) FH_ENULLPTR (event detector internal error - ptr is null)
```

This error means that an internal EEM event detector pointer was null when it should have contained a value.

```
(_cerr_sub_err = 44) FH_EHISTEMPTY (history list is empty)
```

This error means that the history list was empty.

```
(_cerr_sub_err = 45) FH_ESEQNUM (sequence or workset number out of sync)
```

This error means that the event detector sequence or workset number was invalid.

```
(_cerr_sub_err = 54) FH_EFDUNAVAIL (connection to event detector unavailable)
```

This error means that the event detector was unavailable.



## CHAPTER 111

# EEM Utility Tcl Command Extensions

The following conventions are used for the syntax documented on the Tcl command extension pages:

- An optional argument is shown within square brackets, for example:

[type ?]

- A question mark ? represents a variable to be entered.
- Choices between arguments are represented by pipes, for example:

priority low|normal|high



---

**Note** For all EEM Tcl command extensions, if there is an error, the returned Tcl result string contains the error information.

---



---

**Note** Arguments for which no numeric range is specified take an integer from -2147483648 to 2147483647, inclusive.

---

- [appl\\_read](#), on page 2294
- [appl\\_reqinfo](#), on page 2294
- [appl\\_setinfo](#), on page 2295
- [counter\\_modify](#), on page 2296
- [description](#), on page 2297
- [fts\\_get\\_stamp](#), on page 2298
- [register\\_counter](#), on page 2299
- [register\\_timer](#), on page 2300
- [timer\\_arm](#), on page 2302
- [timer\\_cancel](#), on page 2303
- [unregister\\_counter](#), on page 2304

## appl\_read

Reads Embedded Event Manager (EEM) application volatile data. This Tcl command extension provides support for reading EEM application volatile data. EEM application volatile data can be published by a Cisco software process that uses the EEM application publish API. EEM application volatile data cannot be published by an EEM policy.




---

**Note** Currently there are no Cisco software processes that publish application volatile data.

---

### Syntax

```
appl_read name ? length ?
```

### Arguments

name	(Mandatory) Name of the application published string data.
length	(Mandatory) Length of the string data to read. Must be an integer number between 1 and 4294967295, inclusive.

### Result String

```
data %s
```

Where data is the application published string data to be read.

### Set\_cerrno

Yes

```
(_cerr_sub_err = 2) FH_ESYSERR (generic/unknown error from OS/system)
```

This error means that the operating system reported an error. The POSIX errno value that is reported with the error should be used to determine the cause of the operating system error.

```
(_cerr_sub_err = 7) FH_ENOSUCHKEY (could not find key)
```

This error means that the application event detector info key or other ID was not found.

```
(_cerr_sub_err = 9) FH_EMEMORY (insufficient memory for request)
```

This error means that an internal EEM request for memory failed.

## appl\_reqinfo

Retrieves previously saved information from the Embedded Event Manager (EEM). This Tcl command extension provides support for retrieving information from EEM that has been previously saved with a unique



key, which must be specified in order to retrieve the information. Note that retrieving the information deletes it from EEM. It must be resaved if it is to be retrieved again.

### Syntax

```
appl_reqinfo key ?
```

### Arguments

key	(Mandatory) The string key of the data.
-----	-----------------------------------------

### Result String

```
data %s
```

Where data is the application string data to be retrieved.

### Set \_cerrno

Yes

```
(_cerr_sub_err = 2) FH_ESYSERR (generic/unknown error from OS/system)
```

This error means that the operating system reported an error. The POSIX errno value that is reported with the error should be used to determine the cause of the operating system error.

```
(_cerr_sub_err = 7) FH_ENOSUCHKEY (could not find key)
```

This error means that the application event detector info key or other ID was not found.

## appl\_setinfo

Saves information in the Embedded Event Manager (EEM). This Tcl command extension provides support for saving information in the Embedded Event Manager that can be retrieved later by the same policy or by another policy. A unique key must be specified. This key allows the information to be retrieved later.

### Syntax

```
appl_setinfo key ? data ?
```

### Arguments

key	(Mandatory) The string key of the data.
data	(Mandatory) The application string data to save.

### Result String

None

**Set\_cerrno**

Yes

```
(_cerr_sub_err = 2) FH_ESYSERR (generic/unknown error from OS/system)
```

This error means that the operating system reported an error. The POSIX errno value that is reported with the error should be used to determine the cause of the operating system error.

```
(_cerr_sub_err = 8) FH_EDUPLICATEKEY (duplicate appl info key)
```

This error means that the application event detector info key or other ID was a duplicate.

```
(_cerr_sub_err = 9) FH_EMEMORY (insufficient memory for request)
```

This error means that an internal EEM request for memory failed.

```
(_cerr_sub_err = 34) FH_EMAXLEN (maximum length exceeded)
```

This error means that the object length or number exceeded the maximum.

```
(_cerr_sub_err = 43) FH_EBADLENGTH (bad API length)
```

This error means that the API message length was invalid.

## counter\_modify

Modifies a counter value.

**Syntax**

```
counter_modify event_id ? val ? op nop|set|inc|dec
```

**Arguments**

event_id	(Mandatory) The counter event ID returned by the <b>register_counter</b> Tcl command extension. Must be an integer between 0 and 4294967295, inclusive.
val	(Mandatory) <b>Note</b> Mandatory except when the op nop argument value combination is specified. <ul style="list-style-type: none"> <li>• If op is set, this argument represents the counter value that is to be set.</li> <li>• If op is inc, this argument is the value by which to increment the counter.</li> <li>• If op is dec, this argument is the value by which to decrement the counter.</li> </ul>

op	<p>(Mandatory)</p> <ul style="list-style-type: none"> <li>• nop--Retrieves the current counter value.</li> <li>• set--Sets the counter value to the given value.</li> <li>• inc--Increments the counter value by the given value.</li> <li>• dec--Decrements the counter value by the given value.</li> </ul>
----	---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

### Result String

```
val_remain %d
```

Where val\_remain is the current value of the counter.

### Set\_cerrno

Yes

```
(_cerr_sub_err = 2) FH_ESYSERR (generic/unknown error from OS/system)
```

This error means that the operating system reported an error. The POSIX errno value that is reported with the error should be used to determine the cause of the operating system error.

```
(_cerr_sub_err = 11) FH_ENOSUCHESID (unknown event specification ID)
```

This error means that the event specification ID could not be matched when the event was being registered or that an event detector internal event structure is corrupt.

```
(_cerr_sub_err = 22) FH_ENULLPTR (event detector internal error - ptr is null)
```

This error means that an internal EEM event detector pointer was null when it should have contained a value.

```
(_cerr_sub_err = 30) FH_ECTBADOPER (bad counter threshold operator)
```

This error means that the counter event detector set or modify operator was invalid.

## description

Provides a brief description of the registered policy.

### Syntax

```
description ?
```

### Arguments

line	(Optional) Brief description of the policy consisting of 1 to 240 characters.
------	-------------------------------------------------------------------------------

**Result String**

None

**Set\_cerrno**

Yes

**Sample Usage**

The description statement is entered by the author of the policy. It can appear before or after any event registration statement in Tcl. The policy can have only one description.




---

**Note** Registration of a policy with more than one description statement will fail.

---

The following example shows how a brief description is provided for the **event\_register\_syslog** policy:

```
::cisco::eem::description "This Tcl command looks for the word count in syslog messages."
::cisco::eem::event_register_syslog tag 1 ...
::cisco::eem::event_register_snmp_object tag 2 ...
::cisco::eem::trigger {
 ::cisco::eem::correlate event 1 and event 2
 ::cisco::eem::attribute tag 1 occurs 1
 ::cisco::eem::attribute tag 2 occurs 1
}
```

## fts\_get\_stamp

Returns the time period elapsed since the last software boot. Use this Tcl command extension to return the number of nanoseconds since boot in an array "nsec nnnn" where nnnn is the number of nanoseconds.

**Syntax**

```
fts_get_stamp
```

**Arguments**

None

**Result String**

```
nsec %d
```

Where nsec is the number of nanoseconds since boot.

**Set\_cerrno**

No

# register\_counter

Registers a counter and returns a counter event ID. This Tcl command extension is used by a counter publisher to perform this registration before using the event ID to manipulate the counter.

## Syntax

```
register_counter name ?
```

## Arguments

<b>name</b>	(Mandatory) The name of the counter to be manipulated.
-------------	--------------------------------------------------------

## Result String

```
event_id %d
event_spec_id %d
```

Where `event_id` is the counter event ID for the specified counter; it can be used to manipulate the counter by the **unregister\_counter** or **counter\_modify** Tcl command extensions. The `event_spec_id` argument is the event specification ID for the specified counter.

## Set\_cerrno

Yes

```
(_cerr_sub_err = 2) FH_ESYSERR (generic/unknown error from OS/system)
```

This error means that the operating system reported an error. The POSIX `errno` value that is reported with the error should be used to determine the cause of the operating system error.

```
(_cerr_sub_err = 4) FH_EINITONCE (Init() is not yet done, or done twice.)
```

This error means that the request to register the specific event was made before the EEM event detector had completed its initialization.

```
(_cerr_sub_err = 6) FH_EBADEVENTTYPE (unknown EEM event type)
```

This error means that the event type specified in the internal event specification was invalid.

```
(_cerr_sub_err = 9) FH_EMEMORY (insufficient memory for request)
```

This error means that an internal EEM request for memory failed.

```
(_cerr_sub_err = 10) FH_ECORRUPT (internal EEM API context is corrupt)
```

This error means that the internal EEM API context structure is corrupt.

```
(_cerr_sub_err = 11) FH_ENOSUCHESID (unknown event specification ID)
```

This error means that the event specification ID could not be matched when the event was being registered or that an event detector internal event structure is corrupt.

```
(_cerr_sub_err = 12) FH_ENOSUCHEID (unknown event ID)
```

This error means that the event ID could not be matched when the event was being registered or that an event detector internal event structure is corrupt.

```
(_cerr_sub_err = 16) FH_EBADFMPPTR (bad ptr to fh_p data structure)
```

This error means that the context pointer that is used with each EEM API call is incorrect.

```
(_cerr_sub_err = 17) FH_EBADADDRESS (bad API control block address)
```

This error means that a control block address that was passed in the EEM API was incorrect.

```
(_cerr_sub_err = 22) FH_ENULLPTR (event detector internal error - ptr is null)
```

This error means that an internal EEM event detector pointer was null when it should have contained a value.

```
(_cerr_sub_err = 25) FH_ESUBSEXCEED (number of subscribers exceeded)
```

This error means that the number of timer or counter subscribers exceeded the maximum.

```
(_cerr_sub_err = 26) FH_ESUBSIDXINV (invalid subscriber index)
```

This error means that the subscriber index was invalid.

```
(_cerr_sub_err = 54) FH_EFDUNAVAIL (connection to event detector unavailable)
```

This error means that the event detector was unavailable.

```
(_cerr_sub_err = 56) FH_EFDCONNERR (event detector connection error)
```

This error means that the EEM event detector that handles this request is not available.

## register\_timer

Registers a timer and returns a timer event ID. This Tcl command extension is used by a timer publisher to perform this registration before using the event ID to manipulate the timer if it does not use the **event\_register\_timer** command extension to register as a publisher and subscriber.

### Syntax

```
register_timer watchdog|countdown|absolute|cron name ?
```

### Arguments

name	(Mandatory) The name of the timer to be manipulated.
------	------------------------------------------------------

### Result String

```
event_id %u
```

Where `event_id` is the timer event ID for the specified timer (can be used to manipulate the timer by the `timer_arm` or `timer_cancel` command extensions).

### Set `_cerrno`

Yes

```
(_cerr_sub_err = 2) FH_ESYSERR (generic/unknown error from OS/system)
```

This error means that the operating system reported an error. The POSIX `errno` value that is reported with the error should be used to determine the cause of the operating system error.

```
(_cerr_sub_err = 4) FH_EINITONCE (Init() is not yet done, or done twice.)
```

This error means that the request to register the specific event was made before the EEM event detector had completed its initialization.

```
(_cerr_sub_err = 6) FH_EBADEVENTTYPE (unknown EEM event type)
```

This error means that the event type specified in the internal event specification was invalid.

```
(_cerr_sub_err = 9) FH_EMEMORY (insufficient memory for request)
```

This error means that an internal EEM request for memory failed.

```
(_cerr_sub_err = 10) FH_ECORRUPT (internal EEM API context is corrupt)
```

This error means that the internal EEM API context structure is corrupt.

```
(_cerr_sub_err = 11) FH_ENOSUCHESID (unknown event specification ID)
```

This error means that the event specification ID could not be matched when the event was being registered or that an event detector internal event structure is corrupt.

```
(_cerr_sub_err = 16) FH_EBADFMPPTR (bad ptr to fh_p data structure)
```

This error means that the context pointer that is used with each EEM API call is incorrect.

```
(_cerr_sub_err = 17) FH_EBADADDRESS (bad API control block address)
```

This error means that a control block address that was passed in the EEM API was incorrect.

```
(_cerr_sub_err = 22) FH_ENULLPTR (event detector internal error - ptr is null)
```

This error means that an internal EEM event detector pointer was null when it should have contained a value.

```
(_cerr_sub_err = 25) FH_ESUBSEXCEED (number of subscribers exceeded)
```

This error means that the number of timer or counter subscribers exceeded the maximum.

```
(_cerr_sub_err = 26) FH_ESUBSIDXINV (invalid subscriber index)
```

This error means that the subscriber index was invalid.

```
(_cerr_sub_err = 54) FH_EFDUNAVAIL (connection to event detector unavailable)
```

This error means that the event detector was unavailable.

```
(_cerr_sub_err = 56) FH_EFDCONNERR (event detector connection error)
```

This error means that the EEM event detector that handles this request is not available.

## timer\_arm

Arms a timer. The type could be CRON, watchdog, countdown, or absolute.

### Syntax

```
timer_arm event_id ? cron_entry ?|time ?
```

### Arguments

event_id	(Mandatory) The timer event ID returned by the <b>register_timer</b> command extension. Must be an integer between 0 and 4294967295, inclusive.
cron_entry	(Mandatory) Must exist if the timer type is CRON. Must not exist for other types of timer. CRON timer specification uses the format of the CRON table entry.
time	(Mandatory) Must exist if the timer type is not CRON. Must not exist if the timer type is CRON. For watchdog and countdown timers, the number of seconds and milliseconds until the timer expires; for an absolute timer, the calendar time of the expiration time (specified in SSSSSSSSS[.MMM] format, where SSSSSSSSS must be an integer representing seconds between 0 and 4294967295, inclusive, and where MMM must be an integer representing milliseconds between 0 and 999). An absolute expiration date is the number of seconds and milliseconds since January 1, 1970. If the date specified has already passed, the timer expires immediately.

### Result String

```
sec_remain %ld msec_remain %ld
```

Where sec\_remain and msec\_remain are the remaining time before the next expiration of the timer.




---

**Note** A value of 0 will be returned for the sec\_remain and msec\_remain arguments if the timer type is CRON.

---

### Set\_cerrno

Yes

```
(_cerr_sub_err = 2) FH_ESYSERR (generic/unknown error from OS/system)
```

This error means that the operating system reported an error. The POSIX errno value that is reported with the error should be used to determine the cause of the operating system error.



```
(_cerr_sub_err = 6) FH_EBADEVENTTYPE (unknown EEM event type)
```

This error means that the event type specified in the internal event specification was invalid.

```
(_cerr_sub_err = 9) FH_EMEMORY (insufficient memory for request)
```

This error means that an internal EEM request for memory failed.

```
(_cerr_sub_err = 11) FH_ENOSUCHESID (unknown event specification ID)
```

This error means that the event specification ID could not be matched when the event was being registered or that an event detector internal event structure is corrupt.

```
(_cerr_sub_err = 12) FH_ENOSUCHEID (unknown event ID)
```

This error means that the event ID could not be matched when the event was being registered or that an event detector internal event structure is corrupt.

```
(_cerr_sub_err = 22) FH_ENULLPTR (event detector internal error - ptr is null)
```

This error means that an internal EEM event detector pointer was null when it should have contained a value.

```
(_cerr_sub_err = 27) FH_ETMDELAYZR (zero delay time)
```

This error means that the time specified to arm a timer was zero.

```
(_cerr_sub_err = 42) FH_ENOTREGISTERED (request for event spec that is unregistered)
```

This error means that the event was not registered.

```
(_cerr_sub_err = 54) FH_EFDUNAVAIL (connection to event detector unavailable)
```

This error means that the event detector was unavailable.

```
(_cerr_sub_err = 56) FH_EFDCONNERR (event detector connection error)
```

This error means that the EEM event detector that handles this request is not available.

## timer\_cancel

Cancels a timer.

### Syntax

```
timer_cancel event_id ?
```

### Arguments

event_id	(Mandatory) The timer event ID returned by the <b>register_timer</b> command extension. Must be an integer between 0 and 4294967295, inclusive.
----------	-------------------------------------------------------------------------------------------------------------------------------------------------

**Result String**

```
sec_remain %ld msec_remain %ld
```

Where `sec_remain` and `msec_remain` are the remaining time before the next expiration of the timer.




---

**Note** A value of 0 will be returned for `sec_remain` and `msec_remain` if the timer type is CRON .

---

**Set\_cerrno**

Yes

```
(_cerr_sub_err = 2) FH_ESYSERR (generic/unknown error from OS/system)
```

This error means that the operating system reported an error. The POSIX `errno` value that is reported with the error should be used to determine the cause of the operating system error.

```
(_cerr_sub_err = 6) FH_EBADEVENTTYPE (unknown EEM event type)
```

This error means that the event type specified in the internal event specification was invalid.

```
(_cerr_sub_err = 7) FH_ENOSUCHKEY (could not find key)
```

This error means that the application event detector info key or other ID was not found.

```
(_cerr_sub_err = 11) FH_ENOSUCHESID (unknown event specification ID)
```

This error means that the event specification ID could not be matched when the event was being registered or that an event detector internal event structure is corrupt.

```
(_cerr_sub_err = 12) FH_ENOSUCHEID (unknown event ID)
```

This error means that the event ID could not be matched when the event was being registered or that an event detector internal event structure is corrupt.

```
(_cerr_sub_err = 22) FH_ENULLPTR (event detector internal error - ptr is null)
```

This error means that an internal EEM event detector pointer was null when it should have contained a value.

```
(_cerr_sub_err = 54) FH_EFDUNAVAIL (connection to event detector unavailable)
```

This error means that the event detector was unavailable.

```
(_cerr_sub_err = 56) FH_EFDCONNERR (event detector connection error)
```

This error means that the EEM event detector that handles this request is not available.

## unregister\_counter

Unregisters a counter. This Tcl command extension is used by a counter publisher to unregister a counter that was previously registered with the **register\_counter** Tcl command extension.

## Syntax

```
unregister_counter event_id ? event_spec_id ?
```

## Arguments

event_id	(Mandatory) Counter event ID returned by the <b>register_counter</b> command extension. Must be an integer between 0 and 4294967295, inclusive.
event_spec_id	(Mandatory) Counter event specification ID for the specified counter returned by the <b>register_counter</b> command extension. Must be an integer between 0 and 4294967295, inclusive.

## Result String

None

## Set\_cerrno

Yes

```
(_cerr_sub_err = 2) FH_ESYSERR (generic/unknown error from OS/system)
```

This error means that the operating system reported an error. The POSIX errno value that is reported with the error should be used to determine the cause of the operating system error.

```
(_cerr_sub_err = 9) FH_EMEMORY (insufficient memory for request)
```

This error means that an internal EEM request for memory failed.

```
(_cerr_sub_err = 11) FH_ENOSUCHESID (unknown event specification ID)
```

This error means that the event specification ID could not be matched when the event was being registered or that an event detector internal event structure is corrupt.

```
(_cerr_sub_err = 22) FH_ENULLPTR (event detector internal error - ptr is null)
```

This error means that an internal EEM event detector pointer was null when it should have contained a value.

```
(_cerr_sub_err = 26) FH_ESUBSIDXINV (invalid subscriber index)
```

This error means that the subscriber index was invalid.

```
(_cerr_sub_err = 54) FH_EFDUNAVAIL (connection to event detector unavailable)
```

This error means that the event detector was unavailable.

```
(_cerr_sub_err = 56) FH_EFDCONNERR (event detector connection error)
```

This error means that the EEM event detector that handles this request is not available.

■ unregister\_counter



## PART **XVII**

# **Working with the Cisco IOS File System, Configuration Files, and Software Images**

- [Working with the Cisco IOS File System, Configuration Files, and Software Images, on page 2309](#)





## CHAPTER 112

# Working with the Cisco IOS File System, Configuration Files, and Software Images

- [Working with the Flash File System, on page 2309](#)
- [Working with Configuration Files, on page 2318](#)
- [Replacing and Rolling Back Configurations, on page 2330](#)
- [Working with Software Images , on page 2334](#)
- [Copying Image Files Using TFTP, on page 2336](#)
- [Copying Image Files Using FTP, on page 2339](#)
- [Copying Image Files Using RCP, on page 2344](#)
- [Copying an Image File from One Stack Member to Another, on page 2348](#)

## Working with the Flash File System

### Information About the Flash File System

The flash file system is a single flash device on which you can store files. It also provides several commands to help you manage software bundles and configuration files. The default flash file system on the device is named flash:.

As viewed from the active device, or any stack member, flash: refers to the local flash device, which is the device attached to the same device on which the file system is being viewed. In a device stack, each of the flash devices from the various stack members can be viewed from the active device. The names of these flash file systems include the corresponding device member numbers. For example, flash-3:, as viewed from the active device, refers to the same file system as does flash: on stack member 3. Use the **show file systems** privileged EXEC command to list all file systems, including the flash file systems in the device stack.

Only one user at a time can manage the software bundles and configuration files for a device stack .

### Displaying Available File Systems

To display the available file systems on your device, use the **show file systems** privileged EXEC command as shown in this example for a standalone device:

```
Device# show file systems
```

## Displaying Available File Systems

```

File Systems:
 Size(b) Free(b) Type Flags Prefixes
* 15998976 5135872 flash rw flash:
 - - opaque rw bs:
 - - opaque rw vb:
 524288 520138 nvram rw nvram:
 - - network rw tftp:
 - - opaque rw null:
 - - opaque rw system:
 - - opaque ro xmodem:
 - - opaque ro ymodem:

```

This example shows a device stack. In this example, the active device is stack member 1; the file system on stack member 2 is displayed as flash-2; the file system on stack member 3 is displayed as flash-3; and so on up to stack member 8, displayed as flash-8: for a 8-member stack. The example also shows the crashinfo directories and a USB flash drive plugged into the active device:

```

Device# show file systems
File Systems:
 Size(b) Free(b) Type Flags Prefixes
 145898496 5479424 disk rw crashinfo:crashinfo-1:
 248512512 85983232 disk rw crashinfo-2:stby-crashinfo:
 146014208 17301504 disk rw crashinfo-3:
 146014208 0 disk rw crashinfo-4:
 146014208 1572864 disk rw crashinfo-5:
 248512512 30932992 disk rw crashinfo-6:
 146014208 6291456 disk rw crashinfo-7:
 146276352 15728640 disk rw crashinfo-8:
 146276352 73400320 disk rw crashinfo-9:
* 741621760 481730560 disk rw flash:flash-1:
 1622147072 1360527360 disk rw flash-2:stby-flash:
 729546752 469762048 disk rw flash-3:
 729546752 469762048 disk rw flash-4:
 729546752 469762048 disk rw flash-5:
 1622147072 1340604416 disk rw flash-6:
 729546752 469762048 disk rw flash-7:
 1749549056 1487929344 disk rw flash-8:
 1749549056 1487929344 disk rw flash-9:
 0 0 disk rw unix:
 - - disk rw usbflash0:usbflash0-1:
 - - disk rw usbflash0-2: stby-usbflash0:
 - - disk rw usbflash0-3:
 - - disk rw usbflash0-4:
 - - disk rw usbflash0-5:
 - - disk rw usbflash0-6:
 - - disk rw usbflash0-7:
 - - disk rw usbflash0-8:
 - - disk rw usbflash0-9:
 0 0 disk ro webui:
 - - opaque rw system:
 - - opaque rw tmpsys:
 2097152 2055643 nvram rw stby-nvram:
 - - nvram rw stby-rcsf:
 - - opaque rw null:
 - - opaque ro tar:
 - - network rw tftp:
 2097152 2055643 nvram rw nvram:
 - - opaque wo syslog:
 - - network rw rcp:
 - - network rw http:
 - - network rw ftp:
 - - network rw scp:

```



```

- - network rw https:
- - opaque ro cns:
- - opaque rw revrcsf:

```

Table 222: show file systems Field Descriptions

Field	Value
Size(b)	Amount of memory in the file system in bytes.
Free(b)	Amount of free memory in the file system in bytes.
Type	<p>Type of file system.</p> <p><b>disk</b>—The file system is for a flash memory device, USB flash, and crashinfo file.</p> <p><b>network</b>—The file system for network devices; for example, an FTP server or and HTTP server.</p> <p><b>nvr</b>am—The file system is for a NVRAM device.</p> <p><b>opaque</b>—The file system is a locally generated pseudo file system (for example, the system) or a download interface, such as brimux.</p> <p><b>unknown</b>—The file system is an unknown type.</p>
Flags	<p>Permission for file system.</p> <p><b>ro</b>—read-only.</p> <p><b>rw</b>—read/write.</p> <p><b>wo</b>—write-only.</p>

Field	Value
Prefixes	<p>Alias for file system.</p> <p><b>crashinfo:</b>—Crashinfo file.</p> <p><b>flash:</b>—Flash file system.</p> <p><b>ftp:</b>—FTP server.</p> <p><b>http:</b>—HTTP server.</p> <p><b>https:</b>—Secure HTTP server.</p> <p><b>nvr:</b>—NVRAM.</p> <p><b>null:</b>—Null destination for copies. You can copy a remote file to null to find its size.</p> <p><b>r:</b>—Remote Copy Protocol (RCP) server.</p> <p><b>s:</b>—Session Control Protocol (SCP) server.</p> <p><b>system:</b>—Contains the system memory, including the running configuration.</p> <p><b>t:</b>—TFTP network server.</p> <p><b>usbflash0:</b>—USB flash memory.</p> <p><b>x:</b>—Obtain the file from a network machine by using the Xmodem protocol.</p> <p><b>y:</b>—Obtain the file from a network machine by using the Ymodem protocol.</p>

## Setting the Default File System

You can specify the file system or directory that the system uses as the default file system by using the **cd** *filesystem:* privileged EXEC command. You can set the default file system to omit the *filesystem:* argument from related commands. For example, for all privileged EXEC commands that have the optional *filesystem:* argument, the system uses the file system specified by the **cd** command.

By default, the default file system is *flash:*.

You can display the current default file system as specified by the **cd** command by using the **pwd** privileged EXEC command.

## Displaying Information About Files on a File System

You can view a list of the contents of a file system before manipulating its contents. For example, before copying a new configuration file to flash memory, you might want to verify that the file system does not already contain a configuration file with the same name. Similarly, before copying a flash configuration file to another location, you might want to verify its filename for use in another command. To display information about files on a file system, use one of the privileged EXEC commands listed in the following table.

Table 223: Commands for Displaying Information About Files

Command	Description
<b>dir</b> [/all] [filesystem:filename]	Displays a list of files on a file system.
<b>show file systems</b>	Displays more information about each of the files on a file system.
<b>show file information</b> file-url	Displays information about a specific file.
<b>show file descriptors</b>	Displays a list of open file descriptors. File descriptors are the internal representations of open files. You can use this command to see if another user has a file open.

## Changing Directories and Displaying the Working Directory

Follow these steps to change directories and to display the working directory:

### SUMMARY STEPS

1. **enable**
2. **dir** *filesystem:*
3. **cd** *directory\_name*
4. **pwd**
5. **cd**

### DETAILED STEPS

	Command or Action	Purpose
<b>Step 1</b>	<b>enable</b> <b>Example:</b>  Device> <b>enable</b>	Enables privileged EXEC mode.  • Enter your password if prompted.
<b>Step 2</b>	<b>dir</b> <i>filesystem:</i> <b>Example:</b>  Device# <b>dir</b> flash:	Displays the directories on the specified file system. For <i>filesystem:</i> , use flash: for the system board flash device.
<b>Step 3</b>	<b>cd</b> <i>directory_name</i> <b>Example:</b>  Device# <b>cd</b> new_configs	Navigates to the specified directory.  The command example shows how to navigate to the directory named <i>new_configs</i> .
<b>Step 4</b>	<b>pwd</b> <b>Example:</b>	Displays the working directory.

	Command or Action	Purpose
	Device# pwd	
<b>Step 5</b>	<b>cd</b> <b>Example:</b> Device# cd	Navigates to the default directory.

## Creating Directories

Beginning in privileged EXEC mode, follow these steps to create a directory:

### SUMMARY STEPS

1. **dir** *filesystem:*
2. **mkdir** *directory\_name*
3. **dir** *filesystem:*

### DETAILED STEPS

	Command or Action	Purpose
<b>Step 1</b>	<b>dir</b> <i>filesystem:</i> <b>Example:</b> Device# dir flash:	Displays the directories on the specified file system. For <i>filesystem:</i> , use <b>flash:</b> for the system board flash device.
<b>Step 2</b>	<b>mkdir</b> <i>directory_name</i> <b>Example:</b> Device# mkdir new_configs	Creates a new directory. Directory names are case sensitive and are limited to 45 characters between the slashes (/); the name cannot contain control characters, spaces, slashes, quotes, semicolons, or colons.
<b>Step 3</b>	<b>dir</b> <i>filesystem:</i> <b>Example:</b> Device# dir flash:	Verifies your entry.

## Removing Directories

To remove a directory with all its files and subdirectories, use the **delete /force /recursive** *filesystem:/file-url* privileged EXEC command.

Use the **/recursive** keyword to delete the named directory and all subdirectories and the files contained in it. Use the **/force** keyword to suppress the prompting that confirms a deletion of each file in the directory. You are prompted only once at the beginning of this deletion process.

For *filesystem*, use **flash:** for the system board flash device. For *file-url*, enter the name of the directory to be deleted. All of the files in the directory and the directory are removed.



**Caution** When directories are deleted, their contents cannot be recovered.

## Copying Files

To copy a file from a source to a destination, use the **copy source-url destination-url** privileged EXEC command. For the source and destination URLs, you can use **running-config** and **startup-config** keyword shortcuts. For example, the **copy running-config startup-config** command saves the currently running configuration file to the NVRAM section of flash memory to be used as the configuration during system initialization.

You can also copy from special file systems (**xmodem:**, **ymodem:**) as the source for the file from a network machine that uses the Xmodem or Ymodem protocol.

Network file system URLs include ftp:, rcp:, and tftp: and have these syntaxes:

- FTP—ftp:[[/username [:password]@location]/directory]/filename
- RCP—rcp:[[/username@location]/directory]/filename
- TFTP—tftp:[[/location]/directory]/filename

Local writable file systems include flash:

Some invalid combinations of source and destination exist. Specifically, you cannot copy these combinations:

- From a running configuration to a running configuration
- From a startup configuration to a startup configuration
- From a device to the same device (for example, the **copy flash: flash:** command is invalid)

## Copying Files from One Device in a Stack to Another Device in the Same Stack

To copy a file from one device in a stack to another device in the same stack, use the **flash-X:** notation, where **X** is the device number.

To view all devices in a stack, use the **show switch** command in privileged EXEC mode, as in the following example of a 9-member device stack:

```
Switch#show switch
Switch/Stack Mac Address : 046c.9d01.3b80 - Local Mac Address
Mac persistency wait time: 4 mins
```

Switch#	Role	Mac Address	Priority	H/W Version	Current State
*1	Active	046c.9d01.3b80	15	P4B	Ready
2	Standby	046c.9d01.0f80	13	P3C	Ready
3	Member	046c.9d01.1180	11	P4B	Ready
4	Member	046c.9d01.0e80	9	P3C	Ready
5	Member	046c.9d01.4d00	7	P3C	Ready
6	Member	046c.9d01.2800	5	P3C	Ready
7	Member	046c.9d01.6e80	3	P4B	Ready
8	Member	046c.9d01.8180	1	P4B	Ready

To view all file systems available to copy on a specific device, use the **copy** command as in the following example of a 5-member stack:

This example shows how to copy a config file stored in the flash partition of device 2 to the flash partition of device 4. It assumes that device 2 and device 4 are in the same stack.

```
Device# copy flash-2:config.txt flash-4:config.txt
```

## Deleting Files

When you no longer need a file on a flash memory device, you can permanently delete it. To delete a file or directory from a specified flash device, use the **delete** [**/force**] [**/recursive**] [*filesystem:*]/*file-url* privileged EXEC command.

Use the **/recursive** keyword for deleting a directory and all subdirectories and the files contained in it. Use the **/force** keyword to suppress the prompting that confirms a deletion of each file in the directory. You are prompted only once at the beginning of this deletion process. Use the **/force** and **/recursive** keywords for deleting old software images that were installed by using the **archive download-sw** command but are no longer needed.

If you omit the *filesystem:* option, the device uses the default device specified by the **cd** command. For *file-url*, you specify the path (directory) and the name of the file to be deleted.

When you attempt to delete any files, the system prompts you to confirm the deletion.




---

**Caution** When files are deleted, their contents cannot be recovered.

---

This example shows how to delete the file *myconfig* from the default flash memory device:

```
Device# delete myconfig
```

## Creating, Displaying and Extracting Files

You can create a file and write files into it, list the files in a file, and extract the files from a file as described in the next sections.

Beginning in privileged EXEC mode, follow these steps to create a file, display the contents, and extract it:

### SUMMARY STEPS

1. **archive tar /create** *destination-url* **flash:** /*file-url*
2. **archive tar /table** *source-url*
3. **archive tar /xtract** *source-url* **flash:** /*file-url* [*dir/file...*]
4. **more** [ /*ascii* | /*binary* | /*ebcdic* ] /*file-url*

### DETAILED STEPS

	Command or Action	Purpose
Step 1	<b>archive tar /create</b> <i>destination-url</i> <b>flash:</b> / <i>file-url</i> Example:	Creates a file and adds files to it.

	Command or Action	Purpose
	<pre>device# archive tar /create tftp:172.20.10.30/saved. flash:/new-configs</pre>	<p>For destination-url, specify the destination URL alias for the local or network file system and the name of the file to create:</p> <ul style="list-style-type: none"> <li>Local flash file system syntax: <ul style="list-style-type: none"> <li><b>flash:</b></li> </ul> </li> <li>FTP syntax: <ul style="list-style-type: none"> <li><b>ftp:</b>[[//username[:password]@location]/directory]/-filename.</li> </ul> </li> <li>RCP syntax: <ul style="list-style-type: none"> <li><b>rcp:</b>[[//username@location]/directory]/-filename.</li> </ul> </li> <li>TFTP syntax: <ul style="list-style-type: none"> <li><b>tftp:</b>[[//location]/directory]/-filename.</li> </ul> </li> </ul> <p>For <b>flash:/file-url</b>, specify the location on the local flash file system in which the new file is created. You can also specify an optional list of files or directories within the source directory to add to the new file. If none are specified, all files and directories at this level are written to the newly created file.</p>
Step 2	<p><b>archive tar /table source-url</b></p> <p><b>Example:</b></p> <pre>device# archive tar /table flash: /new_configs</pre>	<p>Displays the contents of a file.</p> <p>For <i>source-url</i>, specify the source URL alias for the local or network file system. The <i>-filename</i> is the file to display. These options are supported:</p> <ul style="list-style-type: none"> <li>Local flash file system syntax: <ul style="list-style-type: none"> <li><b>flash:</b></li> </ul> </li> <li>FTP syntax: <ul style="list-style-type: none"> <li><b>ftp:</b>[[//username[:password]@location]/directory]/-filename.</li> </ul> </li> <li>RCP syntax: <ul style="list-style-type: none"> <li><b>rcp:</b>[[//username@location]/directory]/-filename.</li> </ul> </li> <li>TFTP syntax: <ul style="list-style-type: none"> <li><b>tftp:</b>[[//location]/directory]/-filename.</li> </ul> </li> </ul> <p>You can also limit the file displays by specifying a list of files or directories after the file. Only those files appear. If none are specified, all files and directories appear.</p>
Step 3	<p><b>archive tar /xtract source-url flash:/file-url [dir/file...]</b></p> <p><b>Example:</b></p> <pre>device# archive tar /xtract tftp:/172.20.10.30/saved. flash:/new-configs</pre>	<p>Extracts a file into a directory on the flash file system.</p> <p>For <i>source-url</i>, specify the source URL alias for the local file system. The <i>-filename</i> is the file from which to extract files. These options are supported:</p> <ul style="list-style-type: none"> <li>Local flash file system syntax: <ul style="list-style-type: none"> <li><b>flash:</b></li> </ul> </li> <li>FTP syntax:</li> </ul>

	Command or Action	Purpose
		<p><b>ftp</b>:<code>[[//username[:password]@location]/directory]/-filename.</code></p> <ul style="list-style-type: none"> <li>• RCP syntax:</li> </ul> <p><b>rcp</b>:<code>[[//username@location]/directory]/-filename.</code></p> <ul style="list-style-type: none"> <li>• TFTP syntax:</li> </ul> <p><b>tftp</b>:<code>[[//location]/directory]/-filename.</code></p> <p>For <b>flash</b>:<code>/file-url [dir/file...]</code>, specify the location on the local flash file system from which the file is extracted. Use the <code>dir/file...</code> option to specify a list of files or directories within the file to be extracted. If none are specified, all files and directories are extracted.</p>
<b>Step 4</b>	<p><b>more</b> <code>[ /ascii   /binary   /ebcdic ] /file-url</code></p> <p><b>Example:</b></p> <pre>device# more flash:/new-configs</pre>	Displays the contents of any readable file, including a file on a remote file system.

## Working with Configuration Files

### Information on Configuration Files

Configuration files contain commands entered to customize the function of the Cisco IOS software. A way to create a basic configuration file is to use the setup program or to enter the setup privileged EXEC command.

You can copy (download) configuration files from a TFTP, FTP, or RCP server to the running configuration or startup configuration of the switch. You might want to perform this for one of these reasons:

- To restore a backed-up configuration file.
- To use the configuration file for another switch. For example, you might add another switch to your network and want it to have a configuration similar to the original switch. By copying the file to the new switch, you can change the relevant parts rather than recreating the whole file.
- To load the same configuration commands on all the switches in your network so that all the switches have similar configurations.

You can copy (upload) configuration files from the switch to a file server by using TFTP, FTP, or RCP. You might perform this task to back up a current configuration file to a server before changing its contents so that you can later restore the original configuration file from the server.

The protocol you use depends on which type of server you are using. The FTP and RCP transport mechanisms provide faster performance and more reliable delivery of data than TFTP. These improvements are possible because FTP and RCP are built on and use the TCP/IP stack, which is connection-oriented.



## Guidelines for Creating and Using Configuration Files

Creating configuration files can aid in your switch configuration. Configuration files can contain some or all of the commands needed to configure one or more switches. For example, you might want to download the same configuration file to several switches that have the same hardware configuration.

Use these guidelines when creating a configuration file:

- We recommend that you connect through the console port or Ethernet management port for the initial configuration of the switch. If you are accessing the switch through a network connection instead of through a direct connection to the console port or Ethernet management port, keep in mind that some configuration changes (such as changing the switch IP address or disabling ports) can cause a loss of connectivity to the switch.
- If no password has been set on the switch, we recommend that you set one by using the **enable secret** *secret-password* global configuration command.



---

**Note** The **copy {ftp: | rcp: | tftp:} system:running-config** privileged EXEC command loads the configuration files on the switch as if you were entering the commands at the command line. The switch does not erase the existing running configuration before adding the commands. If a command in the copied configuration file replaces a command in the existing configuration file, the existing command is erased. For example, if the copied configuration file contains a different IP address in a particular command than the existing configuration, the IP address in the copied configuration is used. However, some commands in the existing configuration might not be replaced or negated. In this case, the resulting configuration file is a mixture of the existing configuration file and the copied configuration file, with the copied configuration file having precedence.

To restore a configuration file to an exact copy of a file stored on a server, copy the configuration file directly to the startup configuration (by using the **copy {ftp: | rcp: | tftp:} nvram:startup-config** privileged EXEC command), and reload the switch.

---

## Configuration File Types and Location

Startup configuration files are used during system startup to configure the software. Running configuration files contain the current configuration of the software. The two configuration files can be different. For example, you might want to change the configuration for a short time period rather than permanently. In this case, you would change the running configuration but not save the configuration by using the **copy running-config startup-config** privileged EXEC command.

The running configuration is saved in DRAM; the startup configuration is stored in the NVRAM section of flash memory.

## Creating a Configuration File By Using a Text Editor

When creating a configuration file, you must list commands logically so that the system can respond appropriately. This is one method of creating a configuration file:

### SUMMARY STEPS

1. Copy an existing configuration from a switch to a server.

2. Open the configuration file in a text editor, such as vi or emacs on UNIX or Notepad on a PC.
3. Extract the portion of the configuration file with the desired commands, and save it in a new file.
4. Copy the configuration file to the appropriate server location. For example, copy the file to the TFTP directory on the workstation (usually /tftpboot on a UNIX workstation).
5. Make sure the permissions on the file are set to world-read.

## DETAILED STEPS

- 
- Step 1** Copy an existing configuration from a switch to a server.
- Step 2** Open the configuration file in a text editor, such as vi or emacs on UNIX or Notepad on a PC.
- Step 3** Extract the portion of the configuration file with the desired commands, and save it in a new file.
- Step 4** Copy the configuration file to the appropriate server location. For example, copy the file to the TFTP directory on the workstation (usually /tftpboot on a UNIX workstation).
- Step 5** Make sure the permissions on the file are set to world-read.
- 

## Copying Configuration Files By Using TFTP

You can configure the switch by using configuration files you create, download from another switch, or download from a TFTP server. You can copy (upload) configuration files to a TFTP server for storage.

### Preparing to Download or Upload a Configuration File By Using TFTP

Before you begin downloading or uploading a configuration file by using TFTP, do these tasks:

- Ensure that the workstation acting as the TFTP server is properly configured. On a Sun workstation, make sure that the /etc/inetd.conf file contains this line:

```
tftp dgram udp wait root /usr/etc/in.tftpd in.tftpd -p -s /tftpboot
```

Make sure that the /etc/services file contains this line:

```
tftp 69/udp
```




---

**Note** You must restart the inetd daemon after modifying the /etc/inetd.conf and /etc/services files. To restart the daemon, either stop the inetd process and restart it, or enter a **fastboot** command (on the SunOS 4.x) or a **reboot** command (on Solaris 2.x or SunOS 5.x). For more information on the TFTP daemon, see the documentation for your workstation.

---

- Ensure that the switch has a route to the TFTP server. The switch and the TFTP server must be in the same subnetwork if you do not have a router to route traffic between subnets. Check connectivity to the TFTP server by using the **ping** command.
- Ensure that the configuration file to be downloaded is in the correct directory on the TFTP server (usually /tftpboot on a UNIX workstation).
- For download operations, ensure that the permissions on the file are set correctly. The permission on the file should be world-read.

- Before uploading the configuration file, you might need to create an empty file on the TFTP server. To create an empty file, enter the **touch filename** command, where *filename* is the name of the file you will use when uploading it to the server.
- During upload operations, if you are overwriting an existing file (including an empty file, if you had to create one) on the server, ensure that the permissions on the file are set correctly. Permissions on the file should be world-write.

## Downloading the Configuration File By Using TFTP

To configure the switch by using a configuration file downloaded from a TFTP server, follow these steps:

### SUMMARY STEPS

1. Copy the configuration file to the appropriate TFTP directory on the workstation.
2. Verify that the TFTP server is properly configured.
3. Log into the switch through the console port, the Ethernet management port, or a Telnet session.
4. Download the configuration file from the TFTP server to configure the switch.

### DETAILED STEPS

- 
- Step 1** Copy the configuration file to the appropriate TFTP directory on the workstation.
- Step 2** Verify that the TFTP server is properly configured.
- Step 3** Log into the switch through the console port, the Ethernet management port, or a Telnet session.
- Step 4** Download the configuration file from the TFTP server to configure the switch.

Specify the IP address or hostname of the TFTP server and the name of the file to download.

Use one of these privileged EXEC commands:

```
copy tftp:[[//location]/directory]/filename] system:running-config
copy tftp:[[//location]/directory]/filename] nvram:startup-config
copy tftp:[[//location]/directory]/filename] flash[n]:/directory/startup-config
```

The configuration file downloads, and the commands are executed as the file is parsed line-by-line.

---

### Example

This example shows how to configure the software from the file `tokyo-config` at IP address 172.16.2.155:

```
Switch# copy tftp://172.16.2.155/tokyo-config system:running-config
Configure using tokyo-config from 172.16.2.155? [confirm] y
Booting tokyo-config from 172.16.2.155:!!! [OK - 874/16000 bytes]
```

## Uploading the Configuration File By Using TFTP

To upload a configuration file from a switch to a TFTP server for storage, follow these steps:

## SUMMARY STEPS

1. Verify that the TFTP server is properly configured.
2. Log into the switch through the console port, the Ethernet management port, or a Telnet session
3. Upload the switch configuration to the TFTP server. Specify the IP address or hostname of the TFTP server and the destination filename.

## DETAILED STEPS

- 
- Step 1** Verify that the TFTP server is properly configured.
- Step 2** Log into the switch through the console port, the Ethernet management port, or a Telnet session
- Step 3** Upload the switch configuration to the TFTP server. Specify the IP address or hostname of the TFTP server and the destination filename.

Use **one** of these privileged EXEC commands:

- **copy system:running-config tftp:**[[[//location]/directory]/filename]
- **copy nvram:startup-config tftp:**[[[//location]/directory]/filename]
- **copy flash[n]:/directory/startup-config tftp:**[[[//location]/directory]/filename]

The file is uploaded to the TFTP server.

---

### Example

This example shows how to upload a configuration file from a switch to a TFTP server:

```
Switch# copy system:running-config tftp://172.16.2.155/tokyo-config
Write file tokyo-config on host 172.16.2.155? [confirm] y
#
Writing tokyo-config!!! [OK]
```

## Copying a Configuration File from the Device to an FTP Server

You can copy a configuration file from the device to an FTP server.

### Understanding the FTP Username and Password

The FTP protocol requires a client to send a remote username and password on each FTP request to a server. When you copy a configuration file from the device to a server using FTP, the Cisco IOS software sends the first valid username it encounters in the following sequence:

1. The username specified in the **copy** EXEC command, if a username is specified.
2. The username set by the **ip ftp username** global configuration command, if the command is configured.
3. Anonymous.

The device sends the first valid password it encounters in the following sequence:

1. The password specified in the **copy** command, if a password is specified.
2. The password set by the **ip ftp password** command, if the command is configured.
3. The device forms a password *username @devicename.domain*. The variable *username* is the username associated with the current session, *devicename* is the configured host name, and *domain* is the domain of the device.

The username and password must be associated with an account on the FTP server. If you are writing to the server, the FTP server must be properly configured to accept the FTP write request from the user on the device.

If the server has a directory structure, the configuration file or image is written to or copied from the directory associated with the username on the server. For example, if the system image resides in the home directory of a user on the server, specify that user name as the remote username.

Refer to the documentation for your FTP server for more information.

Use the **ip ftp username** and **ip ftp password** global configuration commands to specify a username and password for all copies. Include the username in the **copy** EXEC command if you want to specify a username for that copy operation only.

## Preparing to Download or Upload a Configuration File By Using FTP

Before you begin downloading or uploading a configuration file by using FTP, do these tasks:

- Ensure that the switch has a route to the FTP server. The switch and the FTP server must be in the same subnetwork if you do not have a router to route traffic between subnets. Check connectivity to the FTP server by using the **ping** command.
- If you are accessing the switch through the console or a Telnet session and you do not have a valid username, make sure that the current FTP username is the one that you want to use for the FTP download. You can enter the **show users** privileged EXEC command to view the valid username. If you do not want to use this username, create a new FTP username by using the **ip ftp username** *username* global configuration command during all copy operations. The new username is stored in NVRAM. If you are accessing the switch through a Telnet session and you have a valid username, this username is used, and you do not need to set the FTP username. Include the username in the **copy** command if you want to specify a username for only that copy operation.
- When you upload a configuration file to the FTP server, it must be properly configured to accept the write request from the user on the switch.

For more information, see the documentation for your FTP server.

## Downloading a Configuration File By Using FTP

Beginning in privileged EXEC mode, follow these steps to download a configuration file by using FTP:

### SUMMARY STEPS

1. **configure terminal**
2. **ip ftp username** *username*
3. **ip ftp password** *password*
4. **end**
5. Do one of the following:

- **copy system:running-config ftp:** [[[/[username [:password ]@]location]/directory ]/filename ]
- **copy nvram:startup-config ftp:** [[[/[username [:password ]@]location]/directory ]/filename ]

## DETAILED STEPS

	Command or Action	Purpose
<b>Step 1</b>	<b>configure terminal</b>	Enter global configuration mode on the switch.  This step is required only if you override the default remote username or password (see Steps 2, 3, and 4).
<b>Step 2</b>	<b>ip ftp username</b> <i>username</i>	(Optional) Change the default remote username.
<b>Step 3</b>	<b>ip ftp password</b> <i>password</i>	(Optional) Change the default password.
<b>Step 4</b>	<b>end</b>	Return to privileged EXEC mode.
<b>Step 5</b>	Do one of the following: <ul style="list-style-type: none"> <li>• <b>copy system:running-config ftp:</b> [[[/[username [:password ]@]location]/directory ]/filename ]</li> <li>• <b>copy nvram:startup-config ftp:</b> [[[/[username [:password ]@]location]/directory ]/filename ]</li> </ul>	Using FTP, copy the configuration file from a network server to the running configuration or to the startup configuration file.

### Example

This example shows how to copy a configuration file named `host1-confg` from the `netadmin1` directory on the remote server with an IP address of `172.16.101.101` and to load and run those commands on the switch:

```
Switch# copy ftp://netadmin1:mypass@172.16.101.101/host1-confg
system:running-config
Configure using host1-confg from 172.16.101.101? [confirm]
Connected to 172.16.101.101
Loading 1112 byte file host1-confg:![OK]
Switch#
%SYS-5-CONFIG: Configured from host1-config by ftp from 172.16.101.101
```

This example shows how to specify a remote username of `netadmin1`. The software copies the configuration file `host2-confg` from the `netadmin1` directory on the remote server with an IP address of `172.16.101.101` to the switch startup configuration.

```
Switch# configure terminal
Switch(config)# ip ftp username netadmin1
Switch(config)# ip ftp password mypass
Switch(config)# end
Switch# copy ftp: nvram:startup-config
Address of remote host [255.255.255.255]? 172.16.101.101
Name of configuration file[rtr2-confg]? host2-confg
Configure using host2-confg from 172.16.101.101?[confirm]
Connected to 172.16.101.101
Loading 1112 byte file host2-confg:![OK]
[OK]
Switch#
%SYS-5-CONFIG_NV:Non-volatile store configured from host2-config by ftp from 172.16.101.101
```

## Uploading a Configuration File By Using FTP

Beginning in privileged EXEC mode, follow these steps to upload a configuration file by using FTP:

### SUMMARY STEPS

1. **configure terminal**
2. **ip ftp username** *username*
3. **ip ftp password** *password*
4. **end**
5. Do one of the following:
  - **copy system:running-config ftp:** [[[/[*username* [:*password* ]@]*location*]/*directory* ]/*filename* ]  
or
  - **copy nvram:startup-config ftp:** [[[/[*username* [:*password* ]@]*location*]/*directory* ]/*filename* ]

### DETAILED STEPS

	Command or Action	Purpose
Step 1	<b>configure terminal</b>	Enter global configuration mode on the switch.  This step is required only if you override the default remote username or password (see Steps 2, 3, and 4).
Step 2	<b>ip ftp username</b> <i>username</i>	(Optional) Change the default remote username.
Step 3	<b>ip ftp password</b> <i>password</i>	(Optional) Change the default password.
Step 4	<b>end</b>	Return to privileged EXEC mode.
Step 5	Do one of the following: <ul style="list-style-type: none"> <li>• <b>copy system:running-config ftp:</b> [[[/[<i>username</i> [:<i>password</i> ]@]<i>location</i>]/<i>directory</i> ]/<i>filename</i> ] or</li> <li>• <b>copy nvram:startup-config ftp:</b> [[[/[<i>username</i> [:<i>password</i> ]@]<i>location</i>]/<i>directory</i> ]/<i>filename</i> ]</li> </ul>	Using FTP, store the switch running or startup configuration file to the specified location.

### Example

This example shows how to copy the running configuration file named switch2-config to the netadmin1 directory on the remote host with an IP address of 172.16.101.101:

```
Switch# copy system:running-config
ftp://netadmin1:mypass@172.16.101.101/switch2-config
Write file switch2-config on host 172.16.101.101?[confirm]
Building configuration...[OK]
Connected to 172.16.101.101
Switch#
```

This example shows how to store a startup configuration file on a server by using FTP to copy the file:

```
Switch# configure terminal
Switch(config)# ip ftp username netadmin2
```

```
Switch(config)# ip ftp password mypass
Switch(config)# end
Switch# copy nvram:startup-config ftp:
Remote host[]? 172.16.101.101
Name of configuration file to write [switch2-config]?
Write file switch2-config on host 172.16.101.101?[confirm]
! [OK]
```

## Copying Configuration Files By Using RCP

The RCP provides another method of downloading, uploading, and copying configuration files between remote hosts and the switch. Unlike TFTP, which uses User Datagram Protocol (UDP), a connectionless protocol, RCP uses TCP, which is connection-oriented.

To use RCP to copy files, the server from or to which you will be copying files must support RCP. The RCP copy commands rely on the rsh server (or daemon) on the remote system. To copy files by using RCP, you do not need to create a server for file distribution as you do with TFTP. You only need to have access to a server that supports the remote shell (rsh). (Most UNIX systems support rsh.) Because you are copying a file from one place to another, you must have read permission on the source file and write permission on the destination file. If the destination file does not exist, RCP creates it for you.

The RCP requires a client to send a remote username with each RCP request to a server. When you copy a configuration file from the switch to a server, the Cisco IOS software sends the first valid username in this list:

- The username specified in the **copy** command if a username is specified.
- The username set by the **ip rcmd remote-username username** global configuration command if the command is configured.
- The remote username associated with the current TTY (terminal) process. For example, if the user is connected to the router through Telnet and was authenticated through the **username** command, the switch software sends the Telnet username as the remote username.
- The switch hostname.

For a successful RCP copy request, you must define an account on the network server for the remote username. If the server has a directory structure, the configuration file is written to or copied from the directory associated with the remote username on the server. For example, if the configuration file is in the home directory of a user on the server, specify that user's name as the remote username.

## Preparing to Download or Upload a Configuration File By Using RCP

Before you begin downloading or uploading a configuration file by using RCP, do these tasks:

- Ensure that the workstation acting as the RCP server supports the remote shell (rsh).
- Ensure that the switch has a route to the RCP server. The switch and the server must be in the same subnetwork if you do not have a router to route traffic between subnets. Check connectivity to the RCP server by using the **ping** command.
- If you are accessing the switch through the console or a Telnet session and you do not have a valid username, make sure that the current RCP username is the one that you want to use for the RCP download. You can enter the **show users privileged EXEC** command to view the valid username. If you do not want to use this username, create a new RCP username by using the **ip rcmd remote-username username global**



configuration command to be used during all copy operations. The new username is stored in NVRAM. If you are accessing the switch through a Telnet session and you have a valid username, this username is used, and you do not need to set the RCP username. Include the username in the copy command if you want to specify a username for only that copy operation.

- When you upload a file to the RCP server, it must be properly configured to accept the RCP write request from the user on the switch. For UNIX systems, you must add an entry to the .rhosts file for the remote user on the RCP server. For example, suppose that the switch contains these configuration lines:

```
hostname Switch1
ip rcmd remote-username User0
```

If the switch IP address translates to Switch1.company.com, the .rhosts file for User0 on the RCPserver should contain this line:

```
Switch1.company.com Switch1
```

For more information, see the documentation for your RCP server.

## Downloading a Configuration File By Using RCP

Beginning in privileged EXEC mode, follow these steps to download a configuration file by using RCP:

### SUMMARY STEPS

1. **configure terminal**
2. **ip rcmd remote-username *username***
3. **end**
4. Do one of the following:
  - **copy rcp:[[//*username*@]*location*]/*directory*]/*filename*]**system:running-config****
  - **copy rcp:[[//*username*@]*location*]/*directory*]/*filename*]**nvrाम:startup-config****

### DETAILED STEPS

	Command or Action	Purpose
<b>Step 1</b>	<b>configure terminal</b>	Enter global configuration mode on the switch.  This step is required only if you override the default remote username (see Steps 2 and 3).
<b>Step 2</b>	<b>ip rcmd remote-username <i>username</i></b>	(Optional) Change the default remote username.
<b>Step 3</b>	<b>end</b>	Return to privileged EXEC mode.
<b>Step 4</b>	Do one of the following: <ul style="list-style-type: none"> <li>• <b>copy rcp:[[//<i>username</i>@]<i>location</i>]/<i>directory</i>]/<i>filename</i>]<b>system:running-config</b></b></li> <li>• <b>copy rcp:[[//<i>username</i>@]<i>location</i>]/<i>directory</i>]/<i>filename</i>]<b>nvrाम:startup-config</b></b></li> </ul>	Using RCP, copy the configuration file from a network server to the running configuration or to the startup configuration file.

### Example

This example shows how to copy a configuration file named `host1-config` from the `netadmin1` directory on the remote server with an IP address of `172.16.101.101` and load and run those commands on the switch:

```
Switch# copy rcp://netadmin1@172.16.101.101/host1-config system:running-config
Configure using host1-config from 172.16.101.101? [confirm]
Connected to 172.16.101.101
Loading 1112 byte file host1-config:![OK]
Switch#
%SYS-5-CONFIG: Configured from host1-config by rcp from 172.16.101.101
```

This example shows how to specify a remote username of `netadmin1`. Then it copies the configuration file `host2-config` from the `netadmin1` directory on the remote server with an IP address of `172.16.101.101` to the startup configuration:

```
Switch# configure terminal
Switch(config)# ip rcmd remote-username netadmin1
Switch(config)# end
Switch# copy rcp: nvram:startup-config
Address of remote host [255.255.255.255]? 172.16.101.101
Name of configuration file[rtr2-config]? host2-config
Configure using host2-config from 172.16.101.101?[confirm]
Connected to 172.16.101.101
Loading 1112 byte file host2-config:![OK]
[OK]
Switch#
%SYS-5-CONFIG_NV:Non-volatile store configured from host2-config by rcp from 172.16.101.101
```

## Uploading a Configuration File By Using RCP

Beginning in privileged EXEC mode, follow these steps to upload a configuration file by using RCP

### SUMMARY STEPS

1. **configure terminal**
2. **ip rcmd remote-username *username***
3. **end**
4. Do one of the following:
  - **copy system:running-config rcp:[[*//username@location*]/*directory*]/*filename*]**
  - **copy nvram:startup-config rcp:[[*//username@location*]/*directory*]/*filename*]**

### DETAILED STEPS

	Command or Action	Purpose
Step 1	<b>configure terminal</b>	Enter global configuration mode on the switch.  This step is required only if you override the default remote username (see Steps 2 and 3).
Step 2	<b>ip rcmd remote-username <i>username</i></b>	(Optional) Specify the remote username.

	Command or Action	Purpose
Step 3	end	Return to privileged EXEC mode.
Step 4	Do one of the following: <ul style="list-style-type: none"> <li>• <b>copy system:running-config</b>  <b>rcp:[[/username@]location]/directory]/filename]</b></li> <li>• <b>copy nvram:startup-config</b>  <b>rcp:[[/username@]location]/directory]/filename]</b></li> </ul>	Using RCP, copy the configuration file from a switch running configuration or startup configuration file to a network server.

### Example

This example shows how to copy the running configuration file named switch2-config to the netadmin1 directory on the remote host with an IP address of 172.16.101.101:

```
Switch# copy system:running-config
rcp://netadmin1@172.16.101.101/switch2-config
Write file switch-config on host 172.16.101.101?[confirm]
Building configuration...[OK]
Connected to 172.16.101.101
Switch#
```

This example shows how to store a startup configuration file on a server:

```
Switch# configure terminal
Switch(config)# ip rcmd remote-username netadmin2
Switch(config)# end
Switch# copy nvram:startup-config rcp:
Remote host[]? 172.16.101.101
Name of configuration file to write [switch2-config]?
Write file switch2-config on host 172.16.101.101?[confirm]
![OK]
```

## Clearing Configuration Information

You can clear the configuration information from the startup configuration. If you reboot the switch with no startup configuration, the switch enters the setup program so that you can reconfigure the switch with all new settings.

### Clearing the Startup Configuration File

To clear the contents of your startup configuration, use the **erase nvram:** or the **erase startup-config** privileged EXEC command.



**Note** You cannot restore the startup configuration file after it has been deleted.

### Deleting a Stored Configuration File

To delete a saved configuration from flash memory, use the **delete flash:filename** privileged EXEC command. Depending on the setting of the file prompt global configuration command, you might be prompted for

confirmation before you delete a file. By default, the switch prompts for confirmation on destructive file operations. For more information about the file prompt command, see the Cisco IOS Command Reference for Release 12.4.



---

**Note** You cannot restore a file after it has been deleted.

---

## Replacing and Rolling Back Configurations

The configuration replacement and rollback feature replaces the running configuration with any saved Cisco IOS configuration file. You can use the rollback function to roll back to a previous configuration.

## Information on Configuration Replacement and Rollback

### Configuration Archive

The Cisco IOS configuration archive is intended to provide a mechanism to store, organize, and manage an archive of Cisco IOS configuration files to enhance the configuration rollback capability provided by the **configure replace** command. Before this feature was introduced, you could save copies of the running configuration using the **copy running-config destination-url** command, storing the replacement file either locally or remotely. However, this method lacked any automated file management. On the other hand, the Configuration Replace and Configuration Rollback feature provides the capability to automatically save copies of the running configuration to the Cisco IOS configuration archive. These archived files serve as checkpoint configuration references and can be used by the **configure replace** command to revert to previous configuration states.

The **archive config** command allows you to save Cisco IOS configurations in the configuration archive using a standard location and filename prefix that is automatically appended with an incremental version number (and optional timestamp) as each consecutive file is saved. This functionality provides a means for consistent identification of saved Cisco IOS configuration files. You can specify how many versions of the running configuration are kept in the archive. After the maximum number of files are saved in the archive, the oldest file is automatically deleted when the next, most recent file is saved. The **show archive** command displays information for all configuration files saved in the Cisco IOS configuration archive.

The Cisco IOS configuration archive, in which the configuration files are stored and available for use with the **configure replace** command, can be located on the following file systems: FTP, HTTP, RCP, TFTP.

### Configuration Replace

The **configure replace** privileged EXEC command replaces the running configuration with any saved configuration file. When you enter the **configure replace** command, the running configuration is compared with the specified replacement configuration, and a set of configuration differences is generated. The resulting differences are used to replace the configuration. The configuration replacement operation is usually completed in no more than three passes. To prevent looping behavior no more than five passes are performed.

You can use the **copy source-url running-config** privileged EXEC command to copy a stored configuration file to the running configuration. When using this command as an alternative to the **configure replace target-url** privileged EXEC command, note these major differences:

- The **copy**source-url**running-config** command is a merge operation and preserves all the commands from both the source file and the running configuration. This command does not remove commands from the running configuration that are not present in the source file. In contrast, the **configure replace**target-url command removes commands from the running configuration that are not present in the replacement file and adds commands to the running configuration that are not present.
- You can use a partial configuration file as the source file for the **copy**source-url**running-config** command. You must use a complete configuration file as the replacement file for the **configure replace**target-url command.

## Configuration Rollback

You can also use the **configure replace** command to roll back changes that were made since the previous configuration was saved. Instead of basing the rollback operation on a specific set of changes that were applied, the configuration rollback capability reverts to a specific configuration based on a saved configuration file.

If you want the configuration rollback capability, you must first save the running configuration before making any configuration changes. Then, after entering configuration changes, you can use that saved configuration file to roll back the changes by using the **configure replace**target-url command.

You can specify any saved configuration file as the rollback configuration. You are not limited to a fixed number of rollbacks, as is the case in some rollback models.

## Configuration Guidelines

Follow these guidelines when configuring and performing configuration replacement and rollback:

- Make sure that the switch has free memory larger than the combined size of the two configuration files (the running configuration and the saved replacement configuration). Otherwise, the configuration replacement operation fails.
- Make sure that the switch also has sufficient free memory to execute the configuration replacement or rollback configuration commands.
- Certain configuration commands, such as those pertaining to physical components of a networking device (for example, physical interfaces), cannot be added or removed from the running configuration.
  - A configuration replacement operation cannot remove the **interface**interface-id command line from the running configuration if that interface is physically present on the device.
  - The **interface**interface-id command line cannot be added to the running configuration if no such interface is physically present on the device.
- When using the **configure replace** command, you must specify a saved configuration as the replacement configuration file for the running configuration. The replacement file must be a complete configuration generated by a Cisco IOS device (for example, a configuration generated by the **copy running-config**destination-url command).



---

**Note** If you generate the replacement configuration file externally, it must comply with the format of files generated by Cisco IOS devices.

---

## Configuring the Configuration Archive

Using the **configure replace** command with the configuration archive and with the **archive config** command is optional but offers significant benefit for configuration rollback scenarios. Before using the **archive config** command, you must first configure the configuration archive. Starting in privileged EXEC mode, follow these steps to configure the configuration archive:

### SUMMARY STEPS

1. **configure terminal**
2. **archive**
3. **path***url*
4. **maximum***number*
5. **time-period** *minutes*
6. **end**
7. **show running-config**
8. **copy running-config startup-config**

### DETAILED STEPS

	Command or Action	Purpose
Step 1	<b>configure terminal</b>	Enter global configuration mode.
Step 2	<b>archive</b>	Enter archive configuration mode.
Step 3	<b>path</b> <i>url</i>	Specify the location and filename prefix for the files in the configuration archive
Step 4	<b>maximum</b> <i>number</i>	<p>(Optional) Set the maximum number of archive files of the running configuration to be saved in the configuration archive .</p> <p><i>number</i>-Maximum files of the running configuration file in the configuration archive. Valid values are from 1 to 14. The default is 10.</p> <p><b>Note</b> Before using this command, you must first enter the <b>path</b> archive configuration command to specify the location and filename prefix for the files in the configuration archive.</p>
Step 5	<b>time-period</b> <i>minutes</i>	<p>(Optional) Set the time increment for automatically saving an archive file of the running configuration in the configuration archive.</p> <p><i>minutes</i>-Specify how often, in minutes, to automatically save an archive file of the running configuration in the configuration archive</p>
Step 6	<b>end</b>	Return to privileged EXEC mode.
Step 7	<b>show running-config</b>	Verify the configuration.

	Command or Action	Purpose
Step 8	<code>copy running-config startup-config</code>	(Optional) Save your entries in the configuration file.

## Performing a Configuration Replacement or Rollback Operation

Starting in privileged EXEC mode, follow these steps to replace the running configuration file with a saved configuration file:

### SUMMARY STEPS

1. `archive config`
2. `configure terminal`
3. Make necessary changes to the running configuration.
4. `exit`
5. `configure replace target-url [list] [force] [time seconds] [nolock]`
6. `configure confirm`
7. `copy running-config startup-config`

### DETAILED STEPS

#### Step 1 `archive config`

(Optional) Save the running configuration file to the configuration archive.

**Note** Enter the **path** archive configuration command before using this command.

#### Step 2 `configure terminal`

Enter global configuration mode.

#### Step 3 Make necessary changes to the running configuration.

—

#### Step 4 `exit`

Return to privileged EXEC mode.

#### Step 5 `configure replace target-url [list] [force] [time seconds] [nolock]`

Replace the running configuration file with a saved configuration file.

**target-url**—URL (accessible by the file system) of the saved configuration file that is to replace the running configuration, such as the configuration file created in Step 2 by using the **archive config** privileged EXEC command

**list**—Display a list of the command entries applied by the software parser during each pass of the configuration replacement operation. The total number of passes also appears.

**force**—Replace the running configuration file with the specified saved configuration file without prompting you for confirmation.

**timeseconds**—Specify the time (in seconds) within which you must enter the **configure confirm** command to confirm replacement of the running configuration file. If you do not enter the **configure confirm** command within the specified time limit, the configuration replacement operation is automatically stopped. (In other words, the running configuration file is restored to the configuration that existed before you entered the **configure replace** command).

**Note** You must first enable the configuration archive before you can use the **time** seconds command line option.

**no-lock**— Disable the locking of the running configuration file that prevents other users from changing the running configuration during a configuration replacement operation.

**Step 6** **configure confirm**

(Optional) Confirm replacement of the running configuration with a saved configuration file.

**Note** Use this command only if the **time** seconds keyword and argument of the **configure replace** command are specified.

**Step 7** **copy running-config startup-config**

(Optional) Save your entries in the configuration file.

## Working with Software Images

### Information on Working with Software Images

This section describes how to archive (download and upload) software image files, which contain the system software, the Cisco IOS code, and the embedded device manager software.



**Note** Instead of using the **copy** privileged EXEC command or the **archive tar** privileged EXEC command, we recommend using the **archive download-sw** and **archive upload-sw** privileged EXEC commands to download and upload software image files. For switch stacks, the **archive download-sw** and **archive upload-sw** privileged EXEC commands can only be used through the stack's active switch. Software images downloaded to the stack's active switch are automatically downloaded to the rest of the stack members.

To upgrade a switch in the stack that has an incompatible software image, use the **archive copy-sw** privileged EXEC command to copy the software image from an existing stack member to the incompatible switch. That switch automatically reloads and joins the stack as a fully functioning member.

You can download a switch image file from a TFTP, FTP, or RCP server to upgrade the switch software. If you do not have access to a TFTP server, you can download a software image file directly to your PC or workstation by using a web browser (HTTP) and then by using the device manager or Cisco Network Assistant to upgrade your switch. For information about upgrading your switch by using a TFTP server or a web browser (HTTP), see the release notes.

You can replace the current image with the new one or keep the current image in flash memory after a download.

You upload a switch image file to a TFTP, FTP, or RCP server for backup purposes. You can use this uploaded image for future downloads to the same switch or to another of the same type.

The protocol that you use depends on which type of server you are using. The FTP and RCP transport mechanisms provide faster performance and more reliable delivery of data than TFTP. These improvements are possible because FTP and RCP are built on and use the TCP/IP stack, which is connection-oriented.





**Note** For a list of software images and the supported upgrade paths, see the release notes.

## Image Location on the Switch

The Cisco IOS image is stored as a .bin file in a directory that shows the version number. A subdirectory contains the files needed for web management. The image is stored on the system board flash memory (flash:).

You can use the **show version** privileged EXEC command to see the software version that is currently running on your switch. In the display, check the line that begins with System image file is... . It shows the directory name in flash memory where the image is stored.

You can also use the **dir** filesystem : privileged EXEC command to see the directory names of other software images that might be stored in flash memory.

## File Format of Images on a Server or Cisco.com

Software images located on a server or downloaded from Cisco.com are provided in a tar file format, which contains these files:

- An info file, which serves as a table of contents for the tar file
- One or more subdirectories containing other images and files, such as Cisco IOS images and web management files

This example shows some of the information contained in the info file. The table provides additional details about this information:

```
system_type:0x00000000: image-name
 image_family:xxxx
 stacking_number:x
 info_end:

version_suffix:xxxx
 version_directory:image-name
 image_system_type_id:0x00000000
 image_name:image-nameB.bin
 ios_image_file_size:6398464
 total_image_file_size:8133632
 image_feature:IP|LAYER_3|PLUS|MIN_DRAM_MEG=128
 image_family:xxxx
 stacking_number:x
 board_ids:0x401100c4 0x00000000 0x00000001 0x00000003 0x00000002 0x00008000 0x00008002

0x40110000
 info_end
```

**Table 224: info File Description**

Field	Description
version_suffix	Specifies the Cisco IOS image version string suffix
version_directory	Specifies the directory where the Cisco IOS image and the HTML subdirectory are installed

Field	Description
image_name	Specifies the name of the Cisco IOS image within the tar file
ios_image_file_size	Specifies the Cisco IOS image size in the tar file, which is an approximate measure of how much flash memory is required to hold just the Cisco IOS image
total_image_file_size	Specifies the size of all the images (the Cisco IOS image and the web management files) in the tar file, which is an approximate measure of how much flash memory is required to hold them
image_feature	Describes the core functionality of the image
image_min_dram	Specifies the minimum amount of DRAM needed to run this image
image_family	Describes the family of products on which the software can be installed

## Copying Image Files Using TFTP

You can download a switch image from a TFTP server or upload the image from the switch to a TFTP server.

You download a switch image file from a server to upgrade the switch software. You can overwrite the current image with the new one or keep the current image after a download.

You upload a switch image file to a server for backup purposes; this uploaded image can be used for future downloads to the same or another switch of the same type .



**Note** Instead of using the **copy** privileged EXEC command or the **archive tar** privileged EXEC command, we recommend using the **archive download-sw** and **archive upload-sw** privileged EXEC commands to download and upload software image files. For switch stacks, the **archive download-sw** and **archive upload-sw** privileged EXEC commands can only be used through the stack's active switch. Software images downloaded to the stack's active switch are automatically downloaded to the rest of the stack members.

To upgrade a switch with an incompatible software image, use the **archive copy-sw** privileged EXEC command to copy the software image from an existing stack member to the incompatible switch. That switch automatically reloads and joins the stack as a fully functioning member.

## Preparing to Download or Upload an Image File By Using TFTP

Before you begin downloading or uploading an image file by using TFTP, do these tasks:

- Ensure that the workstation acting as the TFTP server is properly configured. On a Sun workstation, make sure that the `/etc/inetd.conf` file contains this line:

```
tftp dgram udp wait root /usr/etc/in.tftpd in.tftpd -p -s /tftpboot
```

Make sure that the `/etc/services` file contains this line:

```
tftp 69/udp
```



---

**Note** You must restart the `inetd` daemon after modifying the `/etc/inetd.conf` and `/etc/services` files. To restart the daemon, either stop the `inetd` process and restart it, or enter a `fastboot` command (on the SunOS 4.x) or a **reboot** command (on Solaris 2.x or SunOS 5.x). For more information on the TFTP daemon, see the documentation for your workstation.

---

- Ensure that the switch has a route to the TFTP server. The switch and the TFTP server must be in the same subnetwork if you do not have a router to route traffic between subnets. Check connectivity to the TFTP server by using the **ping** command.
- Ensure that the image to be downloaded is in the correct directory on the TFTP server (usually `/tftpboot` on a UNIX workstation).
- For download operations, ensure that the permissions on the file are set correctly. The permission on the file should be world-read.
- Before uploading the image file, you might need to create an empty file on the TFTP server. To create an empty file, enter the **touch** filename command, where filename is the name of the file you will use when uploading the image to the server.
- During upload operations, if you are overwriting an existing file (including an empty file, if you had to create one) on the server, ensure that the permissions on the file are set correctly. Permissions on the file should be world-write.

## Downloading an Image File By Using TFTP

You can download a new image file and replace the current image or keep the current image.

Beginning in privileged EXEC mode, follow Steps 1 through 3 to download a new image from a TFTP server and overwrite the existing image. To keep the current image, go to Step 3.

### SUMMARY STEPS

1. Copy the image to the appropriate TFTP directory on the workstation. Make sure that the TFTP server is properly configured.
2. Log into the switch through the console port or a Telnet session.
3. **archive download-sw /overwrite /reload tftp:** [ [ //location ] /directory ] /image-name.tar
4. **archive download-sw /leave-old-sw /reload tftp:** [ [ //location ] /directory ] /image-name.tar

### DETAILED STEPS

---

**Step 1** Copy the image to the appropriate TFTP directory on the workstation. Make sure that the TFTP server is properly configured.

---

**Step 2** Log into the switch through the console port or a Telnet session.

—

**Step 3** **archive download-sw/overwrite/reload tftp:** [ [ // *location* ] / *directory* ] / *image-name.tar*

Download the image file from the TFTP server to the switch, and overwrite the current image.

- The **/overwrite** option overwrites the software image in flash memory with the downloaded image.
- The **/reload** option reloads the system after downloading the image unless the configuration has been changed and not been saved.
- For // *location* , specify the IP address of the TFTP server.
- For *directory/image-name.tar* specify the directory (optional) and the image to download. Directory and image names are case sensitive.

**Step 4** **archive download-sw/leave-old-sw/reload tftp:** [ [ // *location* ] / *directory* ] / *image-name.tar*

Download the image file from the TFTP server to the switch, and keep the current image.

- The **/leave-old-sw** option keeps the old software version after a download.
- The **/reload** option reloads the system after downloading the image unless the configuration has been changed and not been saved.
- For // *location*, specify the IP address of the TFTP server.
- For *directory/image-name.tar* specify the directory (optional) and the image to download. Directory and image names are case sensitive.

The download algorithm verifies that the image is appropriate for the switch model and that enough DRAM is present, or it cancels the process and reports an error. If you specify the **/overwrite** option, the download algorithm removes the existing image on the flash device whether or not it is the same as the new one, downloads the new image, and then reloads the software.

**Note** If the flash device has sufficient space to hold two images and you want to overwrite one of these images with the same version, you must specify the **/overwrite** option.

If you specify the **/leave-old-sw**, the existing files are not removed. If there is not enough space to install the new image and keep the running image, the download process stops, and an error message is displayed.

The algorithm installs the downloaded image on the system board flash device (flash:). The image is placed into a new directory named with the software version string, and the BOOT environment variable is updated to point to the newly installed image.

If you keep the old image during the download process (you specified the **/leave-old-sw** keyword), you can remove it by entering the **delete/force/recursive filesystem :/ file-url** privileged EXEC command. For *filesystem*, use **flash:** for the system board flash device. For *file-url*, enter the directory name of the old image. All the files in the directory and the directory are removed.

**Note** For the download and upload algorithms to operate properly, do not rename image names

## Uploading an Image File Using TFTP

You can upload an image from the switch to a TFTP server. You can later download this image to the switch or to another switch of the same type.

Use the upload feature only if the web management pages associated with the embedded device manager have been installed with the existing image.

Beginning in privileged EXEC mode, follow these steps to upload an image to a TFTP server:

### SUMMARY STEPS

1. Make sure the TFTP server is properly configured
2. Log into the switch through the console port or a Telnet session.
3. **archive upload-sw tftp:**`[[// location ]/directory ]/image-name .tar`

### DETAILED STEPS

---

**Step 1** Make sure the TFTP server is properly configured

—

**Step 2** Log into the switch through the console port or a Telnet session.

—

**Step 3** **archive upload-sw tftp:**`[[// location ]/directory ]/image-name .tar`

Upload the currently running switch image to the TFTP server.

- For `// location`, specify the IP address of the TFTP server.
- For `/directory/image-name.tar` specify the directory (optional) and the name of the software image to be uploaded. Directory and image names are case sensitive. The `image-name.tar` is the name of the software image to be stored on the server.

The **archive upload-sw** privileged EXEC command builds an image file on the server by uploading these files in order: info, the Cisco IOS image, and the web management files. After these files are uploaded, the upload algorithm creates the tar file format.

**Note** For the download and upload algorithms to operate properly, do not rename image names.

---

## Copying Image Files Using FTP

You can download a switch image from an FTP server or upload the image from the switch to an FTP server.

You download a switch image file from a server to upgrade the switch software. You can overwrite the current image with the new one or keep the current image after a download.

You upload a switch image file to a server for backup purposes. You can use this uploaded image for future downloads to the switch or another switch of the same type.



**Note** Instead of using the **copy** privileged EXEC command or the **archive tar** privileged EXEC command, we recommend using the **archive download-sw** and **archive upload-sw** privileged EXEC commands to download and upload software image files. For switch stacks, the **archive download-sw** and **archive upload-sw** privileged EXEC commands can only be used through the stack's active switch. Software images downloaded to the stack's active switch are automatically downloaded to the rest of the stack members.

To upgrade a switch with an incompatible software image, use the **archive copy-sw** privileged EXEC command to copy the software image from an existing stack member to the incompatible switch. That switch automatically reloads and joins the stack as a fully functioning member.

## Preparing to Download or Upload an Image File By Using FTP

You can copy images files to or from an FTP server.

The FTP protocol requires a client to send a remote username and password on each FTP request to a server. When you copy an image file from the switch to a server by using FTP, the Cisco IOS software sends the first valid username in this list:

- The username specified in the **archive download-sw** or **archive upload-sw** privileged EXEC command if a username is specified.
- The username set by the **ip ftp username** username global configuration command if the command is configured.
- Anonymous.

The switch sends the first valid password in this list:

- The password specified in the **archive download-sw** or **archive upload-sw** privileged EXEC command if a password is specified.
- The password set by the **ip ftp password** password global configuration command if the command is configured.
- The switch forms a password named `username@switchname.domain`. The variable `username` is the username associated with the current session, `switchname` is the configured hostname, and `domain` is the domain of the switch.

The username and password must be associated with an account on the FTP server. If you are writing to the server, the FTP server must be properly configured to accept the FTP write request from you.

Use the **ip ftp username** and **ip ftp password** commands to specify a username and password for all copies. Include the username in the **archive download-sw** or **archive upload-sw** privileged EXEC command if you want to specify a username only for that operation.

If the server has a directory structure, the image file is written to or copied from the directory associated with the username on the server. For example, if the image file resides in the home directory of a user on the server, specify that user's name as the remote username.

Before you begin downloading or uploading an image file by using FTP, do these tasks:

- Ensure that the switch has a route to the FTP server. The switch and the FTP server must be in the same subnetwork if you do not have a router to route traffic between subnets. Check connectivity to the FTP server by using the **ping** command.
- If you are accessing the switch through the console or a Telnet session and you do not have a valid username, make sure that the current FTP username is the one that you want to use for the FTP download. You can enter the **show users** privileged EXEC command to view the valid username. If you do not want to use this username, create a new FTP username by using the **ip ftp username** username global configuration command. This new name will be used during all archive operations. The new username is stored in NVRAM. If you are accessing the switch through a Telnet session and you have a valid username, this username is used, and you do not need to set the FTP username. Include the username in the **archive download-sw** or **archive upload-sw** privileged EXEC command if you want to specify a username for that operation only.
- When you upload an image file to the FTP server, it must be properly configured to accept the write request from the user on the switch.

For more information, see the documentation for your FTP server.

## Downloading an Image File By Using FTP

You can download a new image file and overwrite the current image or keep the current image.

Beginning in privileged EXEC mode, follow Steps 1 through 7 to download a new image from an FTP server and overwrite the existing image. To keep the current image, go to Step 7.

### SUMMARY STEPS

1. Verify that the FTP server is properly configured.
2. Log into the switch through the console port or a Telnet session.
3. **configure terminal**
4. **ip ftp username** *username*
5. **ip ftp password***password*
6. **end**
7. **archive download-sw /overwrite/reload**  
**ftp:** [ [ / / *username* [ :*password* ] @*location* ] / *directory* ] / *image-name.tar*
8. **archive download-sw /leave-old-sw/reload**  
**ftp:** [ [ / / *username* [ :*password* ] @*location* ] / *directory* ] / *image-name.tar*

### DETAILED STEPS

---

**Step 1** Verify that the FTP server is properly configured.

—

**Step 2** Log into the switch through the console port or a Telnet session.

—

**Step 3** **configure terminal**

Enter global configuration mode.

This step is required only if you override the default remote username or password (see Steps 4, 5, and 6).

**Step 4** `ip ftp username username`

(Optional) Change the default remote username.

**Step 5** `ip ftp password password`

(Optional) Change the default password.

**Step 6** `end`

Return to privileged EXEC mode.

**Step 7** `archive download-sw /overwrite/reload ftp: [ [ /username [:password] @location ] /directory ] /image-name.tar`

Download the image file from the FTP server to the switch, and overwrite the current image.

- The `/overwrite` option overwrites the software image in flash memory with the downloaded image.
- The `/reload` option reloads the system after downloading the image unless the configuration has been changed and not been saved.
- For `//username [:password]` specify the username and password; these must be associated with an account on the FTP server.
- For `@ location`, specify the IP address of the FTP server.
- For `directory/image-name.tar`, specify the directory (optional) and the image to download. Directory and image names are case sensitive.

**Step 8** `archive download-sw /leave-old-sw/reload ftp: [ [ /username [:password] @location ] /directory ] /image-name.tar`

Download the image file from the FTP server to the switch, and keep the current image.

- The `/leave-old-sw` option keeps the old software version after a download.
- The `/reload` option reloads the system after downloading the image unless the configuration has been changed and not been saved.
- For `//username [:password]` specify the username and password; these must be associated with an account on the FTP server.
- For `@ location`, specify the IP address of the FTP server.
- For `directory/image-name.tar`, specify the directory (optional) and the image to download. Directory and image names are case sensitive.

The download algorithm verifies that the image is appropriate for the switch model and that enough DRAM is present, or it cancels the process and reports an error. If you specify the `/overwrite` option, the download algorithm removes the existing image on the flash device, whether or not it is the same as the new one, downloads the new image, and then reloads the software.

**Note** If the flash device has sufficient space to hold two images and you want to overwrite one of these images with the same version, you must specify the `/overwrite` option.

If you specify the `/leave-old-sw`, the existing files are not removed. If there is not enough space to install the new image and keep the running image, the download process stops, and an error message is displayed.

The algorithm installs the downloaded image onto the system board flash device (flash:). The image is placed into a new directory named with the software version string, and the BOOT environment variable is updated to point to the newly installed image.

If you kept the old image during the download process (you specified the `/leave-old-sw` keyword), you can remove it by entering the `delete/force/recursive filesystem :/file-url` privileged EXEC command. For `filesystem`, use `flash:` for the system board flash device. For `file-url`, enter the directory name of the old software image. All the files in the directory and the directory are removed.



**Note** For the download and upload algorithms to operate properly, do not rename image names.

## Uploading an Image File By Using FTP

You can upload an image from the switch to an FTP server. You can later download this image to the same switch or to another switch of the same type.

Use the upload feature only if the web management pages associated with the embedded device manager have been installed with the existing image.

Beginning in privileged EXEC mode, follow these steps to upload an image to an FTP server:

### SUMMARY STEPS

1. **configure terminal**
2. **ip ftp username***username*
3. **ip ftp password***password*
4. **end**
5. **archive upload-sw ftp:** [ [ / / [*username* [ :*password*] @ ] *location*] / *directory*] / *image-name.tar*

### DETAILED STEPS

	Command or Action	Purpose
Step 1	<b>configure terminal</b>	Enter global configuration mode.  This step is required only if you override the default remote username or password (see Steps 2, 3, and 4.)
Step 2	<b>ip ftp username</b> <i>username</i>	(Optional) Change the default remote username.
Step 3	<b>ip ftp password</b> <i>password</i>	(Optional) Change the default password.
Step 4	<b>end</b>	Return to privileged EXEC mode.
Step 5	<b>archive upload-sw</b> <b>ftp:</b> [ [ / / [ <i>username</i> [ : <i>password</i> ] @ ] <i>location</i> ] / <i>directory</i> ] / <i>image-name.tar</i>	Upload the currently running switch image to the FTP server.  <ul style="list-style-type: none"> <li>• For <i>//username:password</i>, specify the username and password. These must be associated with an account on the FTP server.</li> <li>• For <i>@location</i>, specify the IP address of the FTP server.</li> <li>• For <i>/directory/image-name.tar</i>, specify the directory (optional) and the name of the software image to be uploaded. Directory and image names are case sensitive. The <i>image-name .tar</i> is the name of the software image to be stored on the server.</li> </ul> <p>The <b>archive upload-sw</b> command builds an image file on the server by uploading these files in order: info, the Cisco IOS image, and the web management files. After these files</p>

	Command or Action	Purpose
		<p>are uploaded, the upload algorithm creates the tar file format.</p> <p><b>Note</b> For the download and upload algorithms to operate properly, do not rename image names.</p>

## Copying Image Files Using RCP

You can download a switch image from an RCP server or upload the image from the switch to an RCP server.

You download a switch image file from a server to upgrade the switch software. You can overwrite the current image with the new one or keep the current image after a download. You upload a switch image file to a server for backup purposes. You can use this uploaded image for future downloads to the same switch or another of the same type.



**Note** Instead of using the **copy** privileged EXEC command or the **archive tar** privileged EXEC command, we recommend using the **archive download-sw** and **archive upload-sw** privileged EXEC commands to download and upload software image files. For switch stacks, the **archive download-sw** and **archive upload-sw** privileged EXEC commands can only be used through the stack's active switch. Software images downloaded to the stack's active switch are automatically downloaded to the rest of the stack members. To upgrade a switch with an incompatible software image, use the **archive copy-sw** privileged EXEC command to copy the software image from an existing stack member to the incompatible switch. That switch automatically reloads and joins the stack as a fully functioning member.

## Preparing to Download or Upload an Image File Using RCP

RCP provides another method of downloading and uploading image files between remote hosts and the switch. Unlike TFTP, which uses User Datagram Protocol (UDP), a connectionless protocol, RCP uses TCP, which is connection-oriented.

To use RCP to copy files, the server from or to which you will be copying files must support RCP. The RCP copy commands rely on the rsh server (or daemon) on the remote system. To copy files by using RCP, you do not need to create a server for file distribution as you do with TFTP. You only need to have access to a server that supports the remote shell (rsh). (Most UNIX systems support rsh.) Because you are copying a file from one place to another, you must have read permission on the source file and write permission on the destination file. If the destination file does not exist, RCP creates it for you.

RCP requires a client to send a remote username on each RCP request to a server. When you copy an image from the switch to a server by using RCP, the Cisco IOS software sends the first valid username in this list:

- The username specified in the **archive download-sw** or **archive upload-sw** privileged EXEC command if a username is specified.
- The username set by the **ip rcmd remote-username** *username* global configuration command if the command is entered.

- The remote username associated with the current TTY (terminal) process. For example, if the user is connected to the router through Telnet and was authenticated through the **username** command, the switch software sends the Telnet username as the remote username.
- The switch hostname.

For the RCP copy request to execute successfully, an account must be defined on the network server for the remote username. If the server has a directory structure, the image file is written to or copied from the directory associated with the remote username on the server. For example, if the image file resides in the home directory of a user on the server, specify that user's name as the remote username.

Before you begin downloading or uploading an image file by using RCP, do these tasks:

- Ensure that the workstation acting as the RCP server supports the remote shell (rsh).
- Ensure that the switch has a route to the RCP server. The switch and the server must be in the same subnetwork if you do not have a router to route traffic between subnets. Check connectivity to the RCP server by using the **ping** command.
- If you are accessing the switch through the console or a Telnet session and you do not have a valid username, make sure that the current RCP username is the one that you want to use for the RCP download. You can enter the **show users** privileged EXEC command to view the valid username. If you do not want to use this username, create a new RCP username by using the **ip rcmd remote-username username** global configuration command to be used during all archive operations. The new username is stored in NVRAM. If you are accessing the switch through a Telnet session and you have a valid username, this username is used, and there is no need to set the RCP username. Include the username in the **archive download-sw** or **archive upload-sw** privileged EXEC command if you want to specify a username only for that operation.
- When you upload an image to the RCP to the server, it must be properly configured to accept the RCP write request from the user on the switch. For UNIX systems, you must add an entry to the .rhosts file for the remote user on the RCP server.

For example, suppose the switch contains these configuration lines:

```
hostname Switch1
ip rcmd remote-username User0
```

If the switch IP address translates to *Switch1.company.com*, the .rhosts file for User0 on the RCP server should contain this line:

```
Switch1.company.com Switch1
```

For more information, see the documentation for your RCP server.

## Downloading an Image File using RCP

You can download a new image file and replace or keep the current image.

Beginning in privileged EXEC mode, follow Steps 1 through 6 to download a new image from an RCP server and overwrite the existing image. To keep the current image, go to Step 6.

### SUMMARY STEPS

1. Verify that the RCP server is properly configured.
2. Log into the switch through the console port or a Telnet session.
3. **configure terminal**
4. **ip rcmd remote-username username**

5. **end**
6. **archive download-sw /overwrite/reload**  
**rcp:** [ [ [ //username@ ] /location ] /directory ] /image-name.tar
7. **archive download-sw /leave-old-sw/reload**  
**rcp:** [ [ [ // [username@] location ] /directory ] /image-name.tar

## DETAILED STEPS

---

- Step 1** Verify that the RCP server is properly configured.
- 
- Step 2** Log into the switch through the console port or a Telnet session.
- 
- Step 3** **configure terminal**
- Enter global configuration mode.
- This step is required only if you override the default remote username or password (see Steps 4, 5, and 6).
- Step 4** **ip rcmd remote-username** *username*
- (Optional) Specify the remote username.
- Step 5** **end**
- Return to privileged EXEC mode.
- Step 6** **archive download-sw /overwrite/reload rcp:** [ [ [ //username@ ] /location ] /directory ] /image-name.tar
- Download the image file from the RCP server to the switch, and overwrite the current image.
- The **/overwrite** option overwrites the software image in flash memory with the downloaded image.
  - The **/reload** option reloads the system after downloading the image unless the configuration has been changed and not been saved.
  - For *//username* specify the username. For the RCP copy request to execute successfully, an account must be defined on the network server for the remote username.
  - For *@ location*, specify the IP address of the RCP server.
  - For */directory/image-name.tar*, specify the directory (optional) and the image to download. Directory and image names are case sensitive.
- Step 7** **archive download-sw /leave-old-sw/reload rcp:** [ [ [ // [username@] location ] /directory ] /image-name.tar
- Download the image file from the FTP server to the switch, and keep the current image.
- The **/leave-old-sw** option keeps the old software version after a download.
  - The **/reload** option reloads the system after downloading the image unless the configuration has been changed and not been saved.
  - For *//username* specify the username. For the RCP copy request to execute, an account must be defined on the network server for the remote username.
  - For *@ location*, specify the IP address of the RCP server.
  - For */directory/image-name.tar*, specify the directory (optional) and the image to download. Directory and image names are case sensitive.

The download algorithm verifies that the image is appropriate for the switch model and that enough DRAM is present, or it cancels the process and reports an error. If you specify the **/overwrite** option, the download algorithm removes the existing image on the flash device, whether or not it is the same as the new one, downloads the new image, and then reloads the software.

**Note** If the flash device has sufficient space to hold two images and you want to overwrite one of these images with the same version, you must specify the **/overwrite** option.

If you specify the **/leave-old-sw**, the existing files are not removed. If there is not enough space to install the new image and keep the running image, the download process stops, and an error message is displayed.

The algorithm installs the downloaded image onto the system board flash device (flash:). The image is placed into a new directory named with the software version string, and the BOOT environment variable is updated to point to the newly installed image.

If you kept the old image during the download process (you specified the **/leave-old-sw** keyword), you can remove it by entering the **delete/force/recursive filesystem :/file-url** privileged EXEC command. For *filesystem*, use **flash:** for the system board flash device. For *file-url*, enter the directory name of the old software image. All the files in the directory and the directory are removed.

**Note** For the download and upload algorithms to operate properly, do not rename image names.

## Uploading an Image File using RCP

You can upload an image from the switch to an RCP server. You can later download this image to the same switch or to another switch of the same type.

The upload feature should be used only if the web management pages associated with the embedded device manager have been installed with the existing image.

Beginning in privileged EXEC mode, follow these steps to upload an image to an RCP server:

### SUMMARY STEPS

1. **configure terminal**
2. **ip rcmd remote-username***username*
3. **end**
4. **archive upload-sw rcp:** [ [ / / [ *username* @ ] *location* ] / *directory* ] / *image-name.tar*

### DETAILED STEPS

	Command or Action	Purpose
Step 1	<b>configure terminal</b>	Enter global configuration mode.  This step is required only if you override the default remote username or password (see Steps 2 and 3.)
Step 2	<b>ip rcmd remote-username</b> <i>username</i>	Optional) Specify the remote username.
Step 3	<b>end</b>	Return to privileged EXEC mode.

	Command or Action	Purpose
Step 4	<b>archive upload-sw</b> <b>rcp:</b> [ [ [ / [ <i>username@</i> ] <i>location</i> ] / <i>directory</i> ] / <i>image-name.tar</i>	<p>Upload the currently running switch image to the RCP server.</p> <ul style="list-style-type: none"> <li>• For <i>//username</i>, specify the username; for the RCP copy request to execute, an account must be defined on the network server for the remote username.</li> <li>• For <i>@location</i>, specify the IP address of the RCP server.</li> <li>• For <i>/directory/image-name.tar</i>, specify the directory (optional) and the name of the software image to be uploaded. Directory and image names are case sensitive.</li> <li>• The <i>image-name.tar</i> is the name of software image to be stored on the server.</li> </ul> <p>The <b>archive upload-sw</b> command builds an image file on the server by uploading these files in order: info, the Cisco IOS image, and the web management files. After these files are uploaded, the upload algorithm creates the tar file format.</p> <p><b>Note</b> For the download and upload algorithms to operate properly, do not rename image names.</p>

## Copying an Image File from One Stack Member to Another

For switch stacks, the **archive download-sw** and **archive upload-sw** privileged EXEC commands can be used only through the stack's active switch. Software images downloaded to the stack's active switch are automatically downloaded to the rest of the stack members.

To upgrade a switch that has an incompatible software image, use the **archive copy-sw** privileged EXEC command to copy the software image from an existing stack member to the one that has incompatible software. That switch automatically reloads and joins the stack as a fully functioning member.



**Note** To successfully use the **archive copy-sw** privileged EXEC command, you must have downloaded from a TFTP server the images for both the stack member switch being added and the stack's active switch. You use the **archive download-sw** privileged EXEC command to perform the download.

Beginning in privileged EXEC mode from the stack member that you want to upgrade, follow these steps to copy the running image file from the flash memory of a different stack member:

### SUMMARY STEPS

1. **archive copy-sw / destination-system** *destination-stack-member-number* / **force-reload***source-stack-member-number*
2. **reload slot***stack-member-number*

## DETAILED STEPS

	Command or Action	Purpose
<b>Step 1</b>	<b>archive copy-sw /destination-system</b> <i>destination-stack-member-number /</i> <b>force-reload</b> <i>source-stack-member-number</i>	<p>Copy the running image file from a stack member, and then unconditionally reload the updated stack member.</p> <p><b>Note</b> At least one stack member must be running the image that is to be copied to the switch that is running the incompatible software</p> <p>For / <b>destination-system</b><i>destination-stack-member-number</i>, specify the number of the stack member (the destination) to which to copy the source running image file. If you do not specify this stack member number, the default is to copy the running image file to all stack members.</p> <p>Specify /<b>force-reload</b> to unconditionally force a system reload after successfully downloading the software image.</p> <p>For <i>source-stack-member-number</i>, specify the number of the stack member (the source) from which to copy the running image file. The stack member number range is 1 to 9.</p>
<b>Step 2</b>	<b>reload slot</b> <i>stack-member-number</i>	Reset the updated stack member, and put this configuration change into effect.

